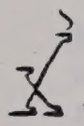
PROJECT XANADU

Box 128, Swarthmore, PA 19081



Electronic Publishing and Hyperstorage™

May 1982

Dear friend,

Thank you for your interest in Project Xanadu*, the book *Literary Machines*, and/or my principal work over the years. They are all essentially one. (If you are one of those people who asked for information some time ago, my great apologies. I, the other project people and our project headquarters have moved several times in the last few years, and many things have been misfiled or mislaid in the trample.)

The book *Literary Machines* — ordering information is given below — details the work of Project Xanadu, which has been the center of my life for over twenty-one years. Project Xanadu is, as you probably know, a computer system seriously intended to become, by various forms of growth, a universal instantaneous publishing system and library, as originally proposed by H.G. Wells and later by Vannevar Bush. Our version will use up-to-date techniques and put the forthcoming hardware (especially the write-once digital "video" disks—) to its best possible use.

The Xanadu Hypertext System* has gone through many stages. I did a number of preliminary versions in the sixties and early seventies with various co-workers, using the computing languages of those times. Since 1976, however, the project has taken several giant steps, and a growing team of very unusual people has been designing and programming the Xanadu Hypertext System for the machines of the eighties. We now believe we've got it. Let me review for you what we have done.

The Xanadu project, described in the book, has created a generalized computer program that essentially grows and assembles into one big electronic library. We conceive of it as a universal publishing system. Let me explain.

Our Xanadu program runs on certain fair-size computers, and manages storage and retrieval in certain special ways. But the program is self-expanding. The same Xanadu program can be run in a network of computers, the same program actually running simultaneously in each machine. In such a setup, a document stored on any one machine is rapidly available to a user on any other.

So far so good; most computer people could set that up. What we have added, however, is the ability to create **links** between documents: any user may attach bookmarks or private notes to any document in the system, as if making marginal notes in a personal copy. Thus each user has an increasingly personalized library that grows and grows without necessarily requiring huge amounts of storage. For you to have "your own" copy of a book, all that is required is one stored copy of the book somewhere on the network, and the small amount of storage space somewhere needed to hold your link information.

But this does not mean a document once stored is "frozen." Its owner may change it continually without invalidating the links that users have attached, since they are permanently attached to the particular place in the changing document. This unique and unusual Xanadu attachment is both to the old version and corresponding places, if any, in more recent versions. Such a powerful facility, we think, must be the heart of any future library network. We believe no one else understands the problem and so no one else has even begun to work on it, whereas we have fully solved it. The problem is **evolving storage with stable links.**

Finally, we plan a system of publication around this which assures all authors (or publishers) of a hassle-free automatic royalty, yet avoids all red tape for other authors who wish to quote or anthologize. Royalty for quoted material automatically goes to the primary owner and that is that. The full publication (or library) system is a totally orderly population of available documents and links with automatic royalty on what is drawn out.

The growth plan is quite straighforward. We will open as a small-scale service storing and supplying on demand, then grow. We will begin public-access service with a single computer, first on straight dial-in and then on Telenet. Additional Xanadu stations will be opened around the country, possibly on a franchise basis, connected by Cylix, ACS or similar high-bandwidth arrangements between stations. Each station will offer local dial-in and storage, but instantaneous pass-through will make it seem that each station contains all documents available.

We do not begin, of course, with "the Library of Congress," but with an empty vessel. We offer a storage service with immediate benefits and immense long-term power. Anyone may store anything, unless it is shown to violate copyright, commit libel or threaten national security. (To the often-asked question "How do you get it in?," the answer is **any way you like**. We will accept ASCII over the phone and presumably floppies though the mail. After we've been open a little while we should be able to afford a Kurzweil scanner.)

The Xanadu system is not a DBMS in the usual sense. You cannot ask for "scans" or searches. It is a literary system for the storage of documents, making explicit the linkage facilities that literature has used from time immemorial. The system is, however, a "decision support system," meaning its contents can be molded to any use. Each viewer may relink, regroup, recategorize, annotate anew or rewrite his permitted universe any way he wants — yet always see the unsmirched original.

I like to think the Xanadu system corresponds best to a naive notion of what computers do. It keeps **track** — of connections, of variants and possibilities and alternatives and of history, with little bother to the user.

The system and its back-end protocol are rather well described in the book. *Literary Machines* is essentially an expanded version of my paper, "Replacing the Printed Word" (Proceedings of the IFIP World Computer Conference, Tokyo and Melbourne, 1980). A related piece, "A New Home for the Mind," appeared in *Datamation* in March of this year. I also included related material in my opening speech at this year's Office Automation conference, and they have asked me to give the featured speech on "the electronic office" at next year's as well.

To the often-asked question, "When?," our standard answer is, and remains, six months. That's when we hope to begin to offer limited service to paying storage-and-linkage customers. But if you want a big in-house system, call for quotation.

The Xanadu Project is alive and well, though our resources are not what we wish they were: programming work continues under the day-to-day direction of Roger Gregory in Ann Arbor, Michigan. The telephone number is 313/663-3637 and the address is Box 7615, Ann Arbor, MI 48107. Demonstrations may be arranged there by special appointment. (Over-the-phone demonstration is impossible because of the need for a tightly-coupled front-end computer to handle user display and interaction. This is not yet exportable.)

While my own special concern as Director of the project is the long-term goal of setting up and marketing the public-access network of Xanadu services, Mr. Gregory is presently handling industrial and commercial applications. (We are quoting big systems to big companies at big prices, especially for code management of large programming projects.)

Fully operational is the indexing system for any number of documents and links between them. Not all of the features described in *Literary Machines* are as yet operational, though all are fully designed. All of these features scale up fully to any size and quantity. (Multi-user and networking are also not as yet implemented, and while they sound tough they are mundane compared to what we have already licked.)

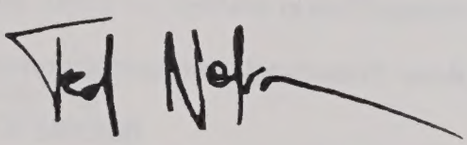
The system runs under the Unix operating system (which must be supplied with a modification license). Unlike conventional programs, the Xanadu program also requires a very large directly-addressable RAM space. Thus the only available machines that will run it as presently written are the VAX, the 68000, the Perkin-Elmer, the Harris, or the National 16000. (However, we are open to custom configuration for other customer machines, especially the 43xx and 33xx series.) Additionally required, of course, is a front-end system — that is, a fairly powerful user terminal — that will provide an easy user interface and translate user requests into the rather intricate back-end protocol.

While interest in our work is considerable throughout the world — especially due to the kind remarks in the 1980 and 1981 *Whole Earth Catalog* — we especially appreciate yours because of your inquiry (or whatever prompted this letter in reply). We stand ready to work with any company that needs our system and knows it. If you recognize the magnitude of what we believe we have done, you will also appreciate our anomalous position. Please understand that while we hope this will become a conventional business enterprise, it is presently an all-out development effort by a somewhat fanatical group of unusual people.

Many people write us with interesting schemes for "networking" and tying the world together by their own proposed methods of document storage, perhaps including you. We urge you to study what we are doing and see if this is not the vehicle you want. These things take a long time to season, and I think we all want to see this thing done right. There may be only one chance.

We continue to need your support and encouragement. (Our well-wishers and believers who are standing by are presently being offered a lifetime user-name of choice, good only when and if the system flies, for $100.)

With very best wishes, I am yours truly,



Theodor H. Nelson, Director

OUR NEXT CONVENTION: Hypercon 8, the Eighth Semi-Annual Xanadu Hypertext Convention, is tentatively scheduled for Boston on Saturday, July 31st (right after Siggraph) at the MIT Artificial Intelligence Laboratory. Unfortunately, we are limited to 100 attendees: prospective attendees should contact us immediately.

AVAILABILITY OF MY BOOKS: *Literary Machines* is available prepaid from me: please send $15 to me at the above address. (Available in bulk or by purchase order from The Distributors, address below.) *The Home Computer Revolution* is $2 (Plus $1.50 postage & handling), from The Distributors, address below. *Computer Lib*, now returned to its 1974 incarnation, is now $12.95 plus $1.50 postage and handling from The Distributors, 702 South Michigan, South Bend, IN 46618.

THEY'RE GOING FAST!

THESE ARE ALREADY TAKEN. Is your name or favorite pseudonym among them?
If it is, Somebody's Already Got It — as a user-name on the Xanadu™ Hypertext System.

47
#
Ah Clem
Alan
Anna
Artificial Department of Intelligence
Bill
Brain Damage
Bush
Centaur
Chip
Citizen
Corvo, Baron
- d -
Deeble
Department of Redundancy Department
Desiderata
Dick
Dionysius
Doc
Dork
Dos Equis
Drexler
Dumbo
- e -
E., Alfred
Elaine
Eric
Erik
Evinrude
Felch
Fiddler
Flitch
Galadriel
Gandalf
Ghu
Gilgamesh
God
Gribble
Griffith
Grubnik
Grump,
Gumbo
Gumby
Hannah
HAHA
Hermes
Herpes
Hillis
Hugh
IFTHENELSE
Igor
Ingerman
It
Jeff
Jonathan
Jove
Key Operator
Kubla
Laura
Leor
Liebowitz
Luball
Luber
Margaret
Mass Driver
Maybe
Mentor
Merc
Metamorph
Midge
Milhous
Minister of Information
Minsky
MSM
Nelsbery
Nero
Nerp, The
No
Nowl
Nonesuch
Orchilla
Pask
Peter
Phlogiston
Pooh
Pope, The
Post
proctologist
Prof
Ridgway
Rob
Roger
Roland
Sarah
Senzig
Shane
Shithead
Sleazy
Sneezy
Snerd
Solomon
Space Man
Spock
Spinx
Student
Steve
Ted
The Director
This is not a
Turoff
Twit
Unicorn
VGLR
Why?
Why, not?
Withau
Wyoming
XCHAR
Wonky, Wonky
Yes
Zeus

We expect the Xanadu system to become a general-purpose public-access storage and publishing utility. That will make it a "network," generally available to subscribers from dial-in or from other networks, such as Arpanet, Tymnet,* The Source,* Unixnet,* etc.

As is the convention in such networks, users will have wide options in choosing their signon names.

We call such user-names "Xandles™," or Xanadu Handles.

Your Xandle may be used as your signon, your billing code, and your address on the system for electronic messages from anyone.

Billing. Xandles are adequate specifications for billing, since our system is sufficiently advanced not to require account numbers.

Locality. Since the Xanadu system will have distributed outlets across the country, such user-names will ordinarily be associated with particular nodes of the network, e.g.

SANTA CLAUS OF CHICAGO

This would be the first person to choose the name "Santa Claus" in the Chicago area. Someone in Cleveland, however, could also have the name "Santa Claus," but he or she would be

SANTA CLAUS OF CLEVELAND

We call such a Xandle a local Xandle, since it is associated with a particular place.

Inter-network addressing. Following the general convention in such matters, it should be possible to send messages to any Xanadu user from any other network by means of the "@" convention, viz.:

SANTA CLAUS OF CLEVELAND @ XANADU

This assumes, of course, that the Xanadu system becomes connected to the general grid of networks within the evolving standard inter-net protocol, which we intend that it will.

These matters explained, we can now propound our Special Offer.

LIMITED TIME OFFER! YOUR VERY OWN LIFETIME GLOBAL XANDLE!

To raise money for our project, we are offering you the official global Xanadu user-name, or Xandle, of your choice, to be usable globally throughout the network with no location suffix, for life. So you can be:

SANTA CLAUS @ XANADU

The only one in the World.

Formally, your hundred-dollar fee is an initiation and registration fee initializing you as a Charter Member of the Xanadu users' community: it gets you a handsome Charter Member certificate (suitable for framing) and wallet card with which to subdue unbelievers. Permission is also granted for you to announce, sport, blazon, spangle, bedizen, brandish and otherwise employ your Xandle in any written, spoken or mumbled form (including but not limited to badges, business cards and letterheads), at any time (including but not limited to Xanadu meetings and independently arranged luaus, soirees and assignations), for the rest of your life. That is the official offer. In addition, when and if the system goes into public operation, your registration fee will be redeemable for one hundred dollars' worth of Xanadu service, and your Xandle will become your official address and account-name if feasible, subject to such rules and regulations of the system as may from time to time come into effect.

As you see, we're not promising all that much. Your Global Xandle is a statement of faith: or if not faith, the symbolic support of high aspiration.

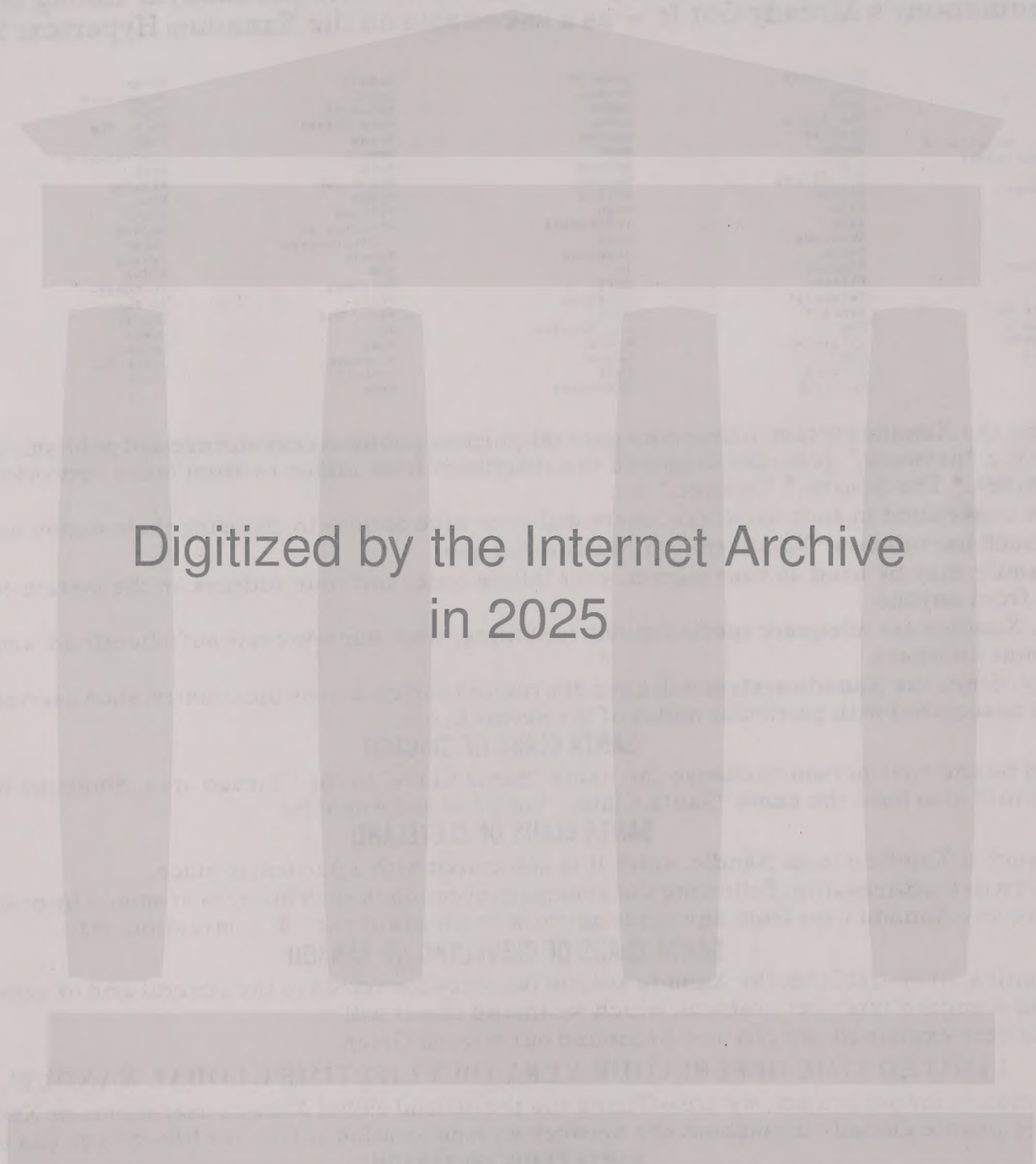
Unpublished Xandles, like unpublished telephone numbers, are allowed. Thus collision avoidance is vital. COLLISIONS BETWEEN PROPOSED XANDLES WILL BE ARBITRATED ARBITRARILY. Capitalization differences are not presently considered sufficient to distinguish Xandles. ASCII punctuation may be included but at least one letter or numeral is presently required.

Note also that trade marks, such as "Pacman," are ordinarily unacceptable. Names of characters in recent fiction (such as "John Galt" and "Luke Skywalker") may have copyright and trademark status and thus be unavailable. Proof of permission from copyright holders may be required. Real names not your own, such as "Richard Nixon," may be similarly encumbered.

Note that we reserve the right to refuse or withdraw a Xandle at any time for any reason, substituting another Xandle mutually satisfactory between us and you.

* Unspecified trade and service marks of various suppliers.

© 1982 Ted Nelson



Digitized by the Internet Archive
in 2025

https://archive.org/details/projectxanadu00tedn

WHIA D2

The computer field is littered with inappropriate terminology. The greatest misnomer is the word "computer" itself, spuriously smearing with numerical connotations what is actually the general-purpose information machine. But almost as juicy a misnomer is the word "videotex(t)," since it stresses a fragile relationship with "video" and doesn't mention the computer at all.

Videotext is essentially a new name for computer-based information services, especially when and if spliced to existing video technology. But videotext is perhaps best understood as among the many public-access computer information services now coming into being.

Unfortunately people seem to be wondering what videotext is, rather than wondering what it should be. We can seize our destiny. The question really is, "What should videotext be?" and that in turn really means, "What should information services be?" Or, to rephrase it one more time, "What do we want in a computer-based information service?

Some of the present vigor with which videotext ideas have been received in the television community might perhaps best be seen as a novice enthusiasm, like wondering in high school about the great questions of existence without being aware that there is such a thing as a professional philosopher.

New enthusiasts for "videotext" may be unaware of the great variety of developments that feed into it-- of the vast and exciting work on interactive personal computing carried on at such centers as Xerox Palo Alto Research Center and the Architecture Machine Group at MIT; of the ferment in software for personal computing (perhaps best observed from the ads in Creative Computing magazine); of the already-existing variety of remote computer services (exemplified by "The Source" and I. P. Sharp's APL services); of the great variety of database vending already underway; or of the remarkable develop of electronic messaging, especially on the Arpanet computer network. Anyone, of course, must be aware of the video game craze, which is now the cutting edge of computer graphics. But I submit that an awareness of all of these is essential before we try to settle on what videotext is going to be.

The greatest danger is that of being sold a story of false necessity, the technical tall tale that videotext "has to be" a certain way.

Another approach often taken is, "What does the public want?" However, in this case the public does not particularly want any-

WNHEAD23

thing, being pretty much unaware of what is possible; this unawareness is to an extent shared with both the video community and much of the computer community.

A third approach to the design of services is to say, "What is the public willing to pay for?" However, this is difficult to estimate, and worse, leads to condescension, partly because you are designing for somebody else, this so-called "public," rather than for the person whose tastes you can most authentically estimate-- yourself. To try to please some "public" that is unlike you is at best like trying to cook for foreigners in their own cuisine.

I would submit, then, that it makes more sense to design a service that will readily please yourself than to try to guess what will please this nameless, formless "public." All the more so in computer services, where there are so many different approaches and the public hasn't seen anything yet.

A SYSTEM ROOTED IN LITERARY EXPERIENCE

At Project Xanadu we are building (to please ourselves) a public-access storage and publishing system-- intended finally as a videotext service-- based essentially on a "literary" model. That is, documents of any size may be published; they may cite and quote each other; and readers may create private notes on them and links to them, just as one may write in the margin of a book.

What the Xanadu™ hypertext system supplies is an instantaneous-access repository for text and graphics that 1) provides any part of any document (or map, photograph, mural, etc.) on demand, 2) has no arbitrary unit boundaries, 3) does not encumber users with complex serial numbers or codes, 4) can grow indefinitely without slowing down markedly in its response time, 5) allows any user to create marginal notes or footnotes to anything (and stores them privately), and 6) allows users to publish their own anthologies or collages of material already in the system-- while 7) assuring royalty payments to copyright owners whenever anything is so used by other publishers.

Linking and windowing among documents-- not scanning-- is the principal form of lookup among them. We conceive of our typical reader as roaming along pre-set links-- mostly published by others -- in an ever-growing corpus.

People ask, quite reasonably, how such a system can be made orderly, and it is indeed our conceptual system of order that is in the long run more important than any proprietary methods we may have. Xanadu hypertext is best understood as a "system of order" for an ever-growing corpus of storage and publication.

Why is this emphasis important? Look at it in reverse. It is easily possible to define a world of innumerable complications, special cases, criss-crossing rules and exceptions. Though some suppliers may temporarily benefit from such arrangements, all lose in the long run. General, all-around use by large numbers of people will only be fostered by extreme simplicity. But it is important to distinguish between simplicity and stupidity. A system can be simple without being coarse or limited. And such simplicity is what I would call elegance.

I would contend that will count most in the design of computer-based information systems for the public will be user simplicity, elegance, power and generality-- not simplicity of manufacture, low manufacturing cost or adaptability to [illegible] arrangements.

Most computer systems are designed backward: starting from a technical base the designer finds comfortable or convenient, decisions and ramifications are successively intertwined until what a user finally sees is a tangle expressing those early decisions. This is wrong. A system should be designed beginning with the performance the user should get-- what we call the system's "virtuality," or apparent structure-- and then working inward and downward to the necessary technical decisions. This procedure is much harder, takes much longer, and is the only right way to design systems for ease of use.

SOME QUESTIONS

I would certainly not want to claim that the system our firm is developing is the ultimate or only right system. But working on it has sensitized us to a number of issues, which I will present here in the form of questions. These questions are intended to foster awareness of the possibilities. I hope that in raising these questions I may also raise interest in, and consciousness of, possible alternatives which have often been ignored. Surely we do not wish to leave to our posterity systems which we have inadvertently limited by fettering the designer's imagination.

WNHIA 025

Why limit yourself to one type of terminal or screen? Some videotext systems have been prematurely specialized to specific viewscreens, presumably in the interest of simplicity. But whether this is justifiable is not clear.

The ostensible connection to "video," in many approaches to videotext, rests upon the presumption of "adapting" the television set. But such "adaptation" is merely a euphemism for hooking the television set to a computer, except in the most limited cases. So the question is, What kind of computer should be allowed?, and the reasonable answer seems to me to be that no kind of computer should not be allowed.

Consider, for instance, that many services are proposed for affluent and well-informed customers. What justification is there for telling such customers that they are restricted to comparatively low-performance screens and terminal computers when they may already possess, or want to possess, high-power personal computers in the Alto/Apollo class ($30,000 U.S.)?

Service ought not be restricted to any one specific type of terminal or user machine, but should optionally feed out to, and interact with, any reasonable programmable device-- provided that the device is prepared to treat the arriving data appropriately: showing text in certain ways, graphics in certain ways, etc.

Why organize on the basis of "pages"? As with screens, some systems have been prematurely specialized to specific paging methods, even requiring numerical codes. Perhaps this kind of design comes from a video way of thinking, especially from the idea of sequential movie and TV titles. The "paging" approach is also strongly suggested (but not required) by conventional simple-minded computer file management systems.

But there is no intrinsic basis whatever for division into pages-- especially if no particular screen hardware is required. Dividing things into arbitrary pages is like scheduling a bus to stop at every crack in the pavement. To put it differently, Why restrict the length of a file or document? Or its entry points or subdivisions?

A related question is, Why not let users define their own screen performance and interaction behaviors? Sophisticated screen-users in the computer community are not agreed on the most desirable styles of viewing for text or graphics. There are many tastes: paging, multi-windowing, scrolling, rolling (smooth

scrolling), and zooming. Different user setups offer different choices among these possibilities; future hardware and firmware will extend these options further.

The different graphical manifestations, and ways of moving and maneuvering in the material, are conceptually, aesthetically and stylistically very different. It would seem to be strangely restrictive to try to tell these sophisticated and obstinate people-- the cutting edge of the customer base-- that they have to use videotext services in one restricted way. It seems to me that the user should be free to move in the material with any screen visualizations and controls he chooses.

Why not take in as well as put out? That is, Why not provide large-scale private storage facilities? If a computer is set up to store material for publication, it is just as easy to store private materials as well, on a large scale. And Why not co-ordinate these private storage services with published material? That is, Why not let a reader make bookmarks and marginal notes, and stand ready to maintain these connections as part of the general indexing facility?

Indeed, Why not let any user be an "information provider"? (Why should the information service be a high-capital market?) There is no requirement for competence or quality in what is provided; why then not let any Tom and Harry publish what they wish? Indeed, Why not make it possible for alternative and dissident materials to be easily made available? This is the Western tradition of freedom of the press-- something which we surely do not propose to relinquish in the name of progress.

And finally, Why not configure our services for indefinite expansion, so as gradually to assimilate all that has been and becomes available? Why not set up to store as much as customers will pay to store? Or to put it most simply, Why not have it all?

WNHIT 027

REFERENCES

Theodor H. Nelson, "The Hypertext." Proceedings of the World Documentation Federation, 1965.

Theodor H. Nelson, Computer Lib. Published by the author; available from The Distributors, 702 South Michigan, South Bend IN 46618.

Theodor H. Nelson, "Replacing the Printed Word." In S. H. Lavington (ed.), Information Processing 80 (Proceedings of the 8th IFIP World Computer Congress), North-Holland Publishing Co., 1980.

Theodor H. Nelson, "Interactive Systems and the Design of Virtuality." Two-part article in Creative Computing, November and December 1980.

Theodor H. Nelson, Literary Machines (in preparation).

WNHJA
(8, Bio)

BIOGRAPHY: THEODOR H. NELSON

Ted Nelson is the director of Project Xanadu, a twenty-year development effort to create a universal publishing system for linked text, graphics and other information. The initial implementation of this system, compatible with videotext approaches, is about to be demonstrated.

A consultant, lecturer and designer, Mr. Nelson is the author of two widely-quoted books, Computer Lib and The Home Computer Revolution, as well as numerous articles. For the year 1980 he was the editor of Creative Computing magazine.

INFORMATION PROCESSING 80, S.H. Lavington (ed.)
North-Holland Publishing Company
IFIP, 1980

Invited Paper

REPLACING THE PRINTED WORD: A COMPLETE LITERARY SYSTEM

Theodor H. NELSON
Project Xanadu, Box 128, Swarthmore, PA 19081, USA

Developments in computer technology now make it possible to store stupendous amounts of written material economically. This leads to radical visions of what may come to be. However, if paper is to be supplanted, it has got to be by something rather like the paper systems we now have, even if in some abstracted and unfamiliar form. What, then, have we? And how abstract and extend it? Our model is "literature"-- defined in a somewhat unusual way.

Under guiding ideas which are not technical but literary, we are implementing a system for storage and retrieval of linked and windowing text. The "document," our fundamental unit, may have windows to any other documents. The evolving corpus is continually expandable without fundamental change. New links and windows may continually add new access pathways to older material. Fast proprietary algorithms render the extreme data fragmentation tolerable in the planned back-end service facility.

THE POSSIBILITIES AND THE PRECONCEPTIONS

Vast text storage, and communication techniques for reaching into it, are a new technological development to be coped with-- which very few people were expecting.

So-called "word processing" systems have already greatly changed the handling of the written word in the office environment; now the question is how to extend this approach. It now is feasible to replace large-scale systems of in-house publishing with computer storage (Lancaster, 1978); and it will soon be practical to do the same for publishing to the open public.

Indeed, true electronic publishing is feasible now-- that is, public-access document systems with digital storage and demand service to the open public.

The problem is not electronic. It is not "software," meaning procedural obstacles to implementation. The problem is conceptual. If such systems are to be promulgated to a wider public-- no longer just in-house-- they must be clear and simple to use, yet offer powerful new features. They may not merely be a clumsy imitation of paper systems.

This is a design problem. It is a problem of creating a conceptual structure, an organizing system of ideas and methods and patterns of work.

There is no universal or obvious approach to this problem, though numerous candidate approaches exist. Various parochial disciplines and ideologies within the computer field, and other fields, have styles of thinking that seem to speak to this problem. How clearly they speak to it is a matter for careful thought.

In word processing, though screen methods for actual edit have become streamlined, all systems appear to bog down at the borders of a document. The conventions of computer storage are little improved, and so secretaries must deal with all the usual file conventions, problems of file naming (and keeping track of names), backup, and keeping backup copies safe and segregated, and so on. Few linkage facilities exist either within or between documents.

A number of on-line communities exist, particularly around such time-sharing systems as the ARPAnet. But they, too, suffer from the conventional problems of file naming and backup, in addition to the greater complexity of using the systems at all. Curiously, the users of such systems still must in most cases give their documents short names, keep paper lists of these names, and juggle backup copies in the same way that secretaries do.

Systems for "electronic mail," though much publicized, are still largely in the telegraphic tradition.

The "office of the future," we are told, is quite near. There seems to be little agreement on what it will be, however. Some say it will have optical scanning for input and paper for output. Some say it will have an "information manager" who will try to keep it all working and keep the system brought up to new revised specifications. All the time. Some say it will not even happen, whatever it is.

Some researchers, impressed by the work of Douglas C. Engelbart and his "NLS" system at Stanford Research Institute, have proceeded on the assumption that good tools for the intellect can be built with good text systems. The NLS system is essentially a community of shared files, with facilities for rapid search and linkage. As implemented it requires extensive training; but Engelbart's ideas have been widely

influential, and would seem to lead toward simpler systems for wider user groups (2,3,4).

A recent development has been the teleconferencing system, most conspicuously developed by Murray Turoff at the New Jersey Institute of Technology. As most often seen, the teleconferencing system is a setup for accumulating messages sequentially from different participants. These messages, always added at the end of the accumulating scroll of text, tend to take on a character not unlike informal conversation. In some versions, however, they have become complicated by tricky sets of rules and an elaborate supervisory function for those people designated as conference leaders.

The field of "information retrieval" appears to have stabilized into a certain basic form. This is usually the storage of a number of items of text which can be scanned on the basis of a formalized user query. (A commercially successful example is the New York Times data bank.) Usually what are stored are article summaries, but this approach could in principle be extended to fulltext as well. However, the training to use the search and query methods of such systems tends to be long and arduous, the cost is high, and users express disgruntlement about the unpredictability of results.

Advocates of artificial intelligence have repeatedly informed us that "everything we want" will be forthcoming in an unspecified manner at an unspecified date in the very near future. However, we are assured that it will involve interactve dialogue with some kind of intelligence whose internal workings and system of thought need not concern us. This Softbeing will understand us perfectly, foresee our needs, wish to keep us completely satisfied, and have nothing to hide.

In summary, I would say that the situation is one of total confusion and odd preoccupations. I see no obvious, let alone conceivable, way that all these concerns and obsessions can be comprised into a single outlook, let alone a common system. (Nevertheless, there are respected researchers, such as J.C.R. Licklider, who seem to think they can (5).)

Perhaps the apparent complication and mutual intractability of these different approaches is related to something else: they all have rather little relation to our present uses of paper.

INTERACTIVE SYSTEMS AND THE DESIGN OF VIRTUALITY

Our approach to a computer design we call "the design of virtuality." By virtuality we mean the seeming of an object or system, its conceptual structure, its atmospherics and its feel.

Every object has a virtuality, a seeming. Natural objects are more or less what they seem to be; man-made objects are not. The virtuality of a house, or an automobile, is what the designer made it-- the structure and qualities that were chosen, and the techniques by which they were realized.

The closest analogy to the design of interactive computer systems, I think, is the making of movies. What counts is effects, not techniques. We are not concerned with just how a certain effect is to be achieved, so much as with what effect is wanted.

An effect is something intended to take place in the mind. Suppose the movie effect desired is a sense of a monster approaching. This can be done by showing a man in a lizard suit-- yaargh-- or animated puppets, or by showing the fright of a person who sees the monster. In other words, a variety of techniques may be selected toward a common effect.

The design of an interactive computer environment, similarly, should not be based on particular hardware, or a particular display device, or a programming technique. It should be based on the intended effect in the mind and heart of the viewer. ("Heart" here is added because we are too seldom mindful of the emotional component in a user's reaction.)

Another way of saying this is that the "systems analysis" for an interactive system should deal with the mental space of the user's experience.

The process is a cycle: study, and design. First we must study the approximate structure of whatever we are designing, and roughly what it is about. Then we design, that is, look to see how the computer's capacities may be made to assume a similar conceptual shape. We will return to the system design process, and its dialectical character, later in this article.

There is one other key constraint in system design: concectual simplicity. If any but highly-trained people are to use a system, it must be extremely simple. It must be simpler by far than anything computer people are accustomed to designing-- a factor of ten, let us say, simpler than what a computer hacker considers "simple."

Popular lore in the computer field holds that simple systems are not "powerful"-- where powerful seems to mean "allowing concise macro-language programming." (This is evidently the view of those who consider TECO a powerful text editor, or, indeed, a text editor.) We believe that true power, meaning easy and focussed control by the user on what he means to do, is not merely compatible with simplicity, it requires it.

This has been a preface to understanding our design. Our approach to electronic text has been to look for the hidden nature of writing as a whole, and the way it is used, to find a paradigm for this design. The virtuality we have designed reflects this endeavor. (In point

of historical fact, this philosophy of virtuality and simplicity arose in parallel with the development of the design, but we think we now know how to design any other system on the same basis.)

THE LITERARY PARADIGM

A piece of writing-- say, a sheet of typed paper on the table-- looks alone and independent. This is quite misleading. Solitary it may be, but it is probably also part of a literature.

By "a literature" we do not mean anything to do with belles-lettres or leather-bound books. We mean it in the same broad sense of "the scientific literature," or that graduate-school question, "Have you looked at the literature?"

A literature is a system of interconnected writings. We do not offer this as our definition, but as a discovered fact. And almost all writing is part of a literature.

The way people write is based in large part on these interconnections.

A person reads an article. He says to himself, "Where have I seen something like that before? Oh, yes--" and the previous connection is brought mentally into play.

Consider how it works in science. A genetic theorist, say, reads current writings in the journals. These refer back, explicitly, to other writings; if he chooses to question the sources, or review their meaning, he is following links as he gets the book and refers to it. He may correspond with colleagues, mentioning in his letters what he has read, and receiving replies suggesting that he read other things. (Again, the letters are implicitly connected to these other writings by implicit links.) Seeking to refresh his ideas, he goes back to Darwin, and also derives inspiration from other things he reads-- the Bible, science fiction. These are linked to his work in his mind.

In his own writing he quotes and cites the things he has read. (Again, explicit links are being made.) Other readers, taking interest in his sources, read them (following the links).

In our Western cultural tradition, writings in principle remain continuously available-- both as recently quoted, and in their original inviolable incarnations-- in a great procession.

So far we have stressed some of the processes of referral and linkage. But also of great importance are controversy and disagreement and reevaluation.

Everyone argues over the interpretation of former writings, even our geneticists. One author will cite (or link to) a passage in Darwin to prove Darwin thought one thing, another will find another passage to try to prove he thought another.

And views of a field, and the way a field's own past is viewed within it, change. A formerly forgotten researcher may come to light (like Mendel), or a highly respected researcher may be discredited (Cyril Burt). And so it goes, on and on. The past is continually changing-- or at least seems to be, as we view it.

There is no predicting the use future people will make of what is written. Any summary, any particular view, is exactly that: the perspective of a particular individual (or school of thought) at a particular time. We cannot know how things will seen in the future. We can assume there will never be a final and definitive view of anything.

And yet this system functions.

LITERATURE IS DEBUGGED.

In other words, even though in every field there is an ever-changing flux of emphasis and perspective and distortion, and an ever-changing fashion in content and approach, the ongoing mechanism of written and published text furnishes a flexible vehicle for this change, continually adapting. Linkage structure between documents forms a flux of invisible threads and rubber bands that hold the thoughts together.

Linkage structure and its ramifications are surprisingly similar in the world of business.

A business letter will say, "In reply to your letter of the 13th..." Or a business form, another key communication, may say in effect, "In response to your order of the 24th of last month, we can supply only half of what you have asked for, but can fill the rest of the order with such-and-such item from our catalog." All of these citations may be thought of as cross-linkages among documents.

The point is clear, whether in science or business or belles lettres. Within bodies of writing, everywhere, there are linkages we tend not to see. The individual document, at hand, is what we deal with; we do not see the total linked collection of them all at once. But they are there, the documents not present as well as those that are, and the grand cat's-cradle among them all.

From this fundamental insight, we of Project Xanadu have endeavored to create a system for text editing and retrieval that will receive, and handle, and present, documents with links between them. We believe there is something very right about the existing system of literature; indeed we suspect that there are things right about it that we don't even know, as with Nature. And so we have tried to mirror, and replicate, and extend, existing literary structure as we have here described it.

But as we said earlier, the design of virtuality is a dialectic. First we must study what there is; then we must venture a design in response to what is. And that design is not dictated by what we have seen, any more than the design of a house is dictated by a hillside. There are only hints.

THE SYSTEM DESIGN

By the "design" of the system we mean its conceptual architecture: the basic ideas of it. But with many systems the general system of concepts is filtered through innumerable complications and accidental features. In this system the basic ideas are the only ideas.

Our approach, then, is to consider all documents as interconnected, or potentially interconnected. Not just documents, but linkages as well, are the fundamental elements of the system. (Links are actually parts of documents, as will be seen later.) It is the different types of links that give both power and complication to the system.

Our system, the Xanadu* Hypertext System, is intended to store a body of writings as an interconnected whole, with linkages, and to provide instantaneous access to any writings within that body.

For this system we have formulated a basic linkage structure that we think meets all the needs of literature as we understand it, including business literature.

Beyond this general idea there is little further to expound, except the link structure and the document conventions, which are all the user really has to learn. Since the network is essentially invisible, all we have to discuss are these links and documents.

LINKS AND COMPOUND OBJECTS

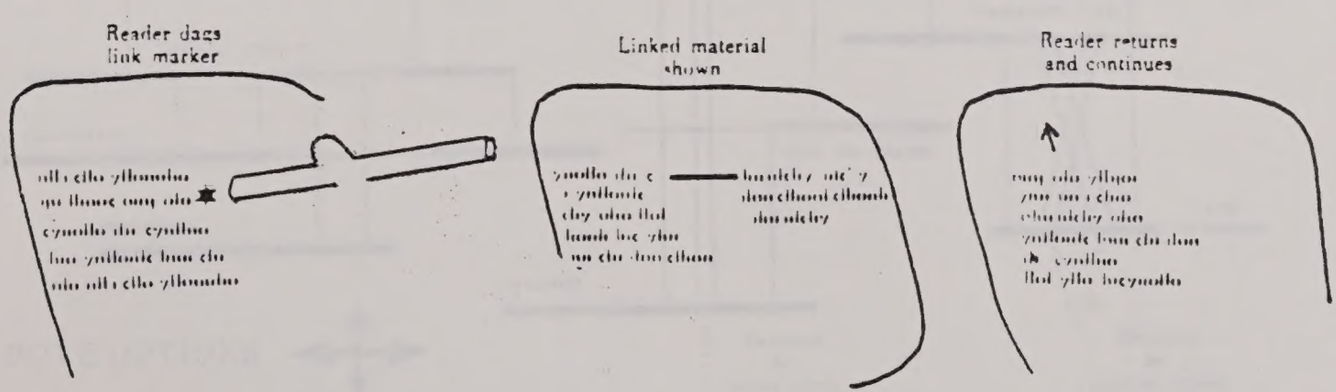
A link is simply a connection between parts of text. It is put in by a human.

A user, on contemplating any two pieces of text, may make a link between them. Thereafter, when he displays either piece of text, and asks to see the links, a link-symbol is displayed, and the other attached text-- if he wishes to see it.

---

* "Xanadu" is a trade and service mark for computer products and services offered by Project Xanadu, Swarthmore, PA.

Figure 1. JUMPING ON A LINK



PARALLEL LINKED TEXT

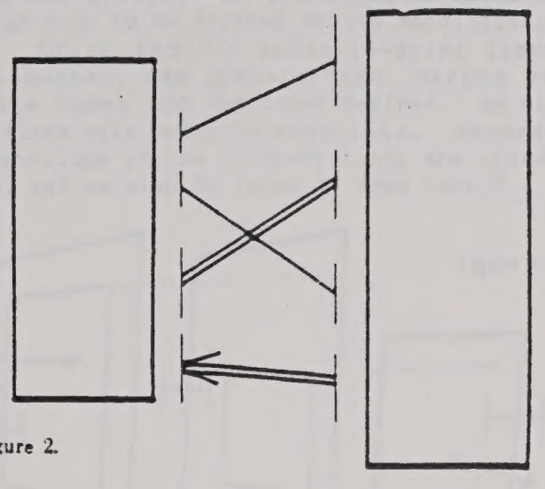
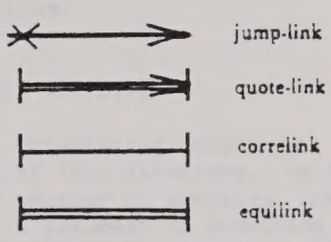


Figure 2.

THE SPECIAL LINKS

This is essentially the basic system. However, its extensions and ramifications, and the unfolding link-types and their capabilities, hold many non-obvious challenges. Certain link types, which we call the "literary set," are the main ones. This is because they have direct analogies in existing literature as everyone knows it.

Figure 3. THE LITERARY LINKS



They are (1) the jump-link, corresponding to the footnote; (2) the quote-link, corresponding to the printed quotation; and (3) the correlink, resembling the marginal note, or "corresponding part." A fourth type, the equilink, will be discussed a little later.

The jump-link is essentially a pointer to a related item, something like a footnote. A reader, after satisfying himself by reading the related item, may push a "return" button to return to the main text.

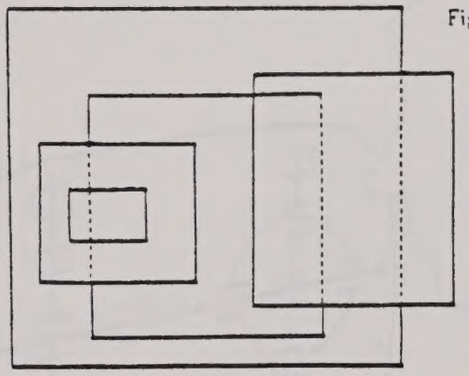
The quote-link makes it possible for a piece of text to be in two places at one time, but with the document of origin still master of the quoted material. (Quote-linkss are automatically maintained by the system, and automatically preserved through a variety of editorial changes, even those involving fragmentation.)

The correlink places a segment of one document "next to" a segment of another document. (Unlike quote-links, correlinks are put in by the user, and in a sense resemble index-tabs or highlighting, being markers to accentuate structural relationships. While correlinks survive fragmentation, it is with some difficulties.)

Correlinks may be used for annotations, or to mark correspondences of any kind. They may also be used to connect parts which correspond to other parts or wholes, as a title corresponds to its document.

COMPOUND DOCUMENT-CLUSTER

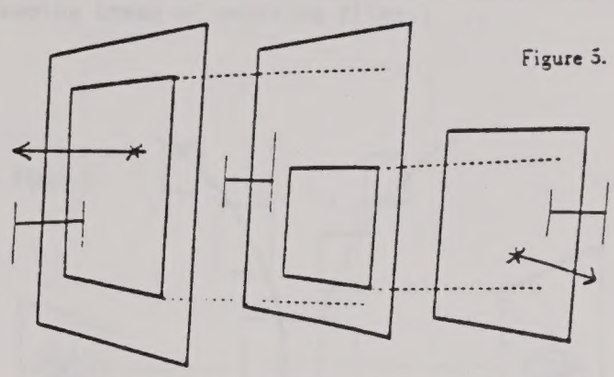
Figure 4.



These structures may of course nest. This makes possible compound documents to any remove, where one document links to another, and so on. One document, embracing another, takes it into itself.

(Many other types of link are allowed within the system. In principle we allow any types of link to be defined by the sophisticated user. These include point-to-point links, point-to-span, and span-to-span, having any separate names and functions desired. We also allow links with multiple endpoints. However, the functions of the literary links are utterly unique, and we wish to focus on them here.)

Figure 5.



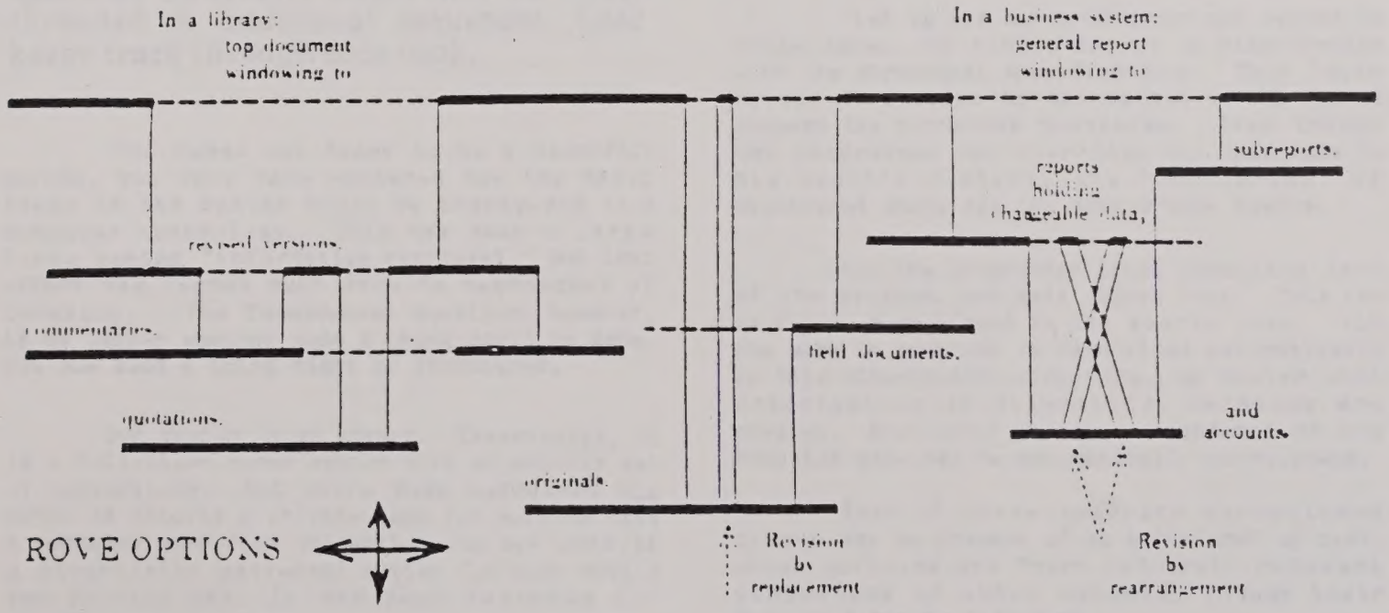
COMPOUND NESTED LINKS

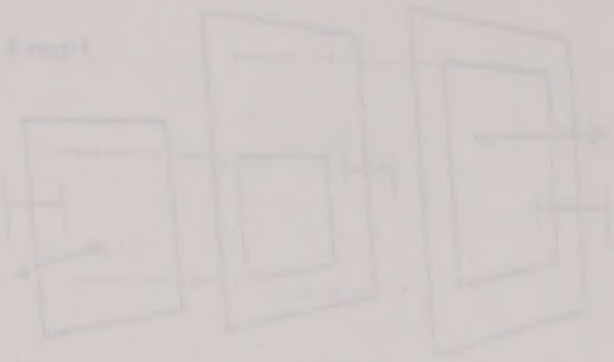
It should be noted that quote-link and correlink are unusually privileged, in that they may applied to parts and to other links to any remove. They are also extensible to all other forms of computer-stored information: graphics, musical scores, and any other form of symbolic materials, whose interrelationships may be shown by these links.

THE DOCUMENT CONVENTION

An interesting choice has been made in the design of this structure. We call a thing a "document" whether it contains text, or links, or both. A document is something designated by a person to be a document, containing text and/or links that he has created. The links are part of the document, though whatever else they link to may not be.

Figure 6. LAYERS OF WINDOWING TEXT. Each horizontal line is a document.





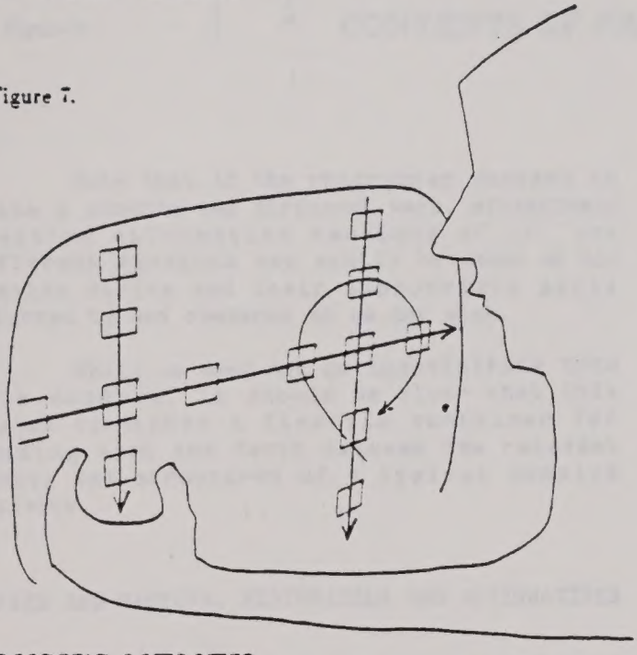
By this convention, then, everything in the system is a document and has an owner. No free-floating materials exist.

Through the same document conventions, the compound structures mentioned earlier maintain the same conventions of integrity and ownership. But one document can be built upon another, and yet another document can be built upon that one, each having links to what was already in place.

ORDERLINESS, EXTENSIBILITY AND GENERALITY

Many have admired Vannevar Bush's postulated "memex" of 1945, but most have supposed out of hand that it was impossible. Bush, it may be recalled, spoke of a hypothetical console that would hold all the documents a person ever read or wrote, and allow him to study "trails" of documents, and put in new trails himself.

Figure 7.



BUSH'S MEMEX:
documents are stored in "trails" of unspecified structure, which may be cross-threaded in additional sequences. User keeps track through code book.

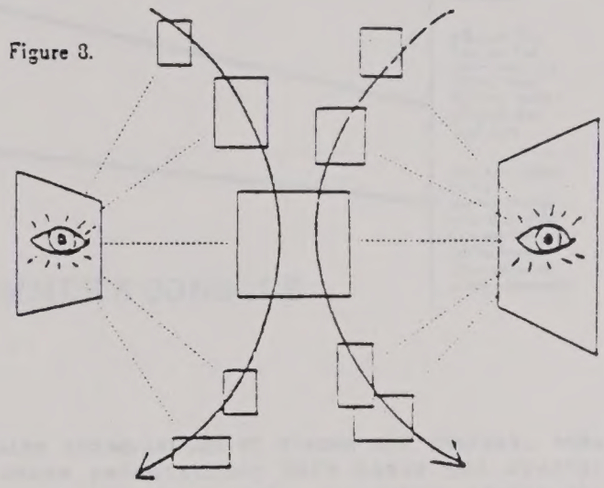
The memex was meant to be a microfilm device, but many have wondered how the basic ideas in its design could be transposed to a computer technology. This has been a large force behind "information retrieval," but that effort has turned much more to techniques of indexing. The fundamental question, however, is no longer whether such a thing could be done, but how such a thing might be structured.

Our system is an answer. Essentially, it is a full-blown memex system with an orderly set of conventions. But where Bush conceived his memex as chiefly a private desk for working with a personal microfilm collection, we see ours as a potentially universal system for both public and private use. It has been designed for indefinite expansion.

Because of this contemplated scale, the system obviously has to be extremely orderly in concept. This we think we have achieved.

The system seems to offer power for the ordering of the most complex mental problems with a minimum of complication and distraction. (At the same time, as already mentioned, it abolishes many of the trivial complexities of keeping track of evolving files.)

Figure 8.



GENERALIZED MEMEX:

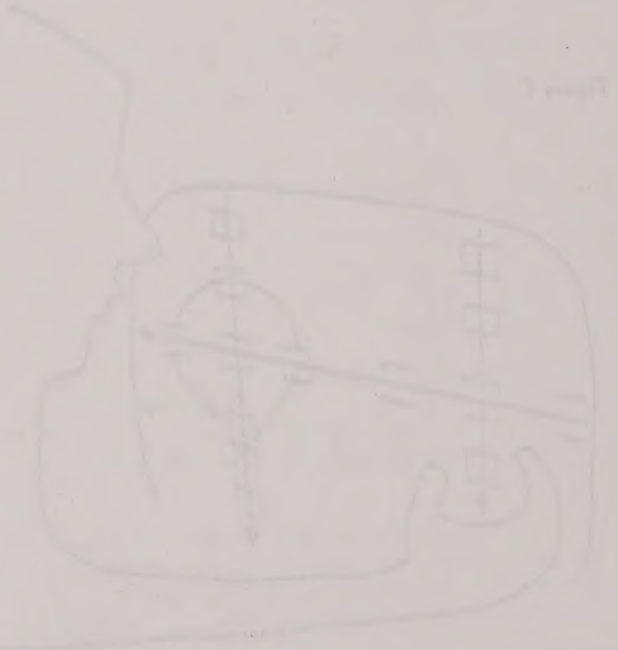
trails are themselves documents.

As an example, let us consider using it as a programmer's console. Suppose that a programmer begins by with a written description of the program he intends to write. (Sometimes this is called a functional specification.) This description is entered into the system. He then creates a detailed breakdown of the parts this program is to have; this description too is entered into the system, and correlinked to the functional specification.

Let us say our programmer now begins to write code. He simply enters it side-by-side with the structural specification. This "side by side" relation is maintained by the system through the correlink mechanism. Even though the programmer may rearrange his materials to his heart's content, the "besideness" is maintained among all the appropriate pieces.

Now the programmer tries compiling part of the program, and gets object code. This too is filed, correlinked to the source code. All the work in progress is maintained automatically in this side-by-side structure, no matter what organization or alternative versions are created. Even error messages engendered by the compiled code may be automatically correlinked.

Each of these separate correlinked pieces may be thought of as a "column" of text, whose portions are "next to" their relevant companions in other columns, though their sequences can be different.



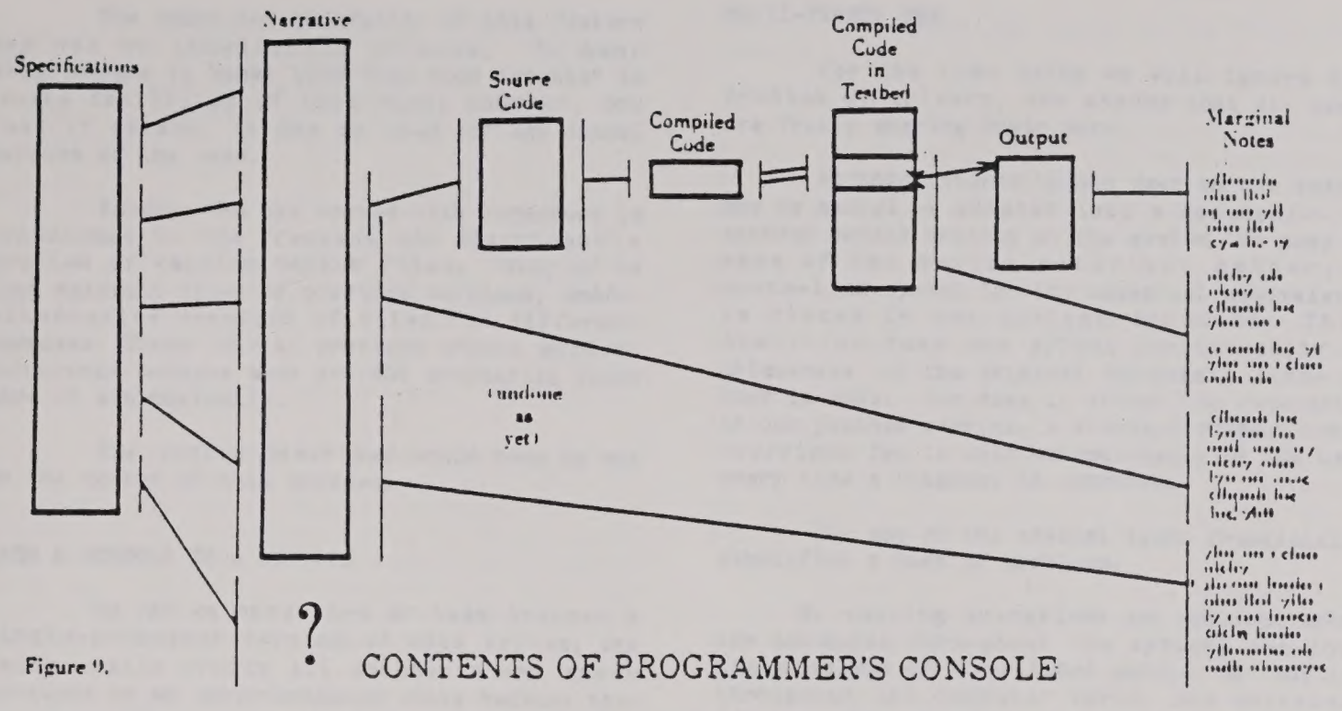


Figure 9. CONTENTS OF PROGRAMMER'S CONSOLE

Note that if the programmer chooses to write a program two different ways, effectively creating alternative versions of it, the different versions may neatly be shown on his display device and their appropriate parts referred to and compared as he may wish.

While we need not go indefinitely into this example, it should be clear that this design furnishes a flexible container for reading back and forth between the relevant levels and structures of a typical complex activity.

COPIES AND BACKUPS, HISTORICALS AND ALTERNATIVES

Our system does not merely handle a document as text and links. Rather, it stores the document canonically as a system of evolving and alternative versions, instantly constructed as needed from the stored fragments, pointers and lists of our unusual data structure. Thus there is no "main" version of a thing, just the ongoing accumulation of pieces and changes, some of whose permutations have names and special linkages. In other words, our system treats all versions of a document as views extracted from the same aggregated object.

The alternative versions employ one other link-type, not discussed earlier, which our data structure uses for alternative and historical versions.

The correspondence between the same thing in two versions of a document is kept track of automatically. An abstract link stretches between these two parts.

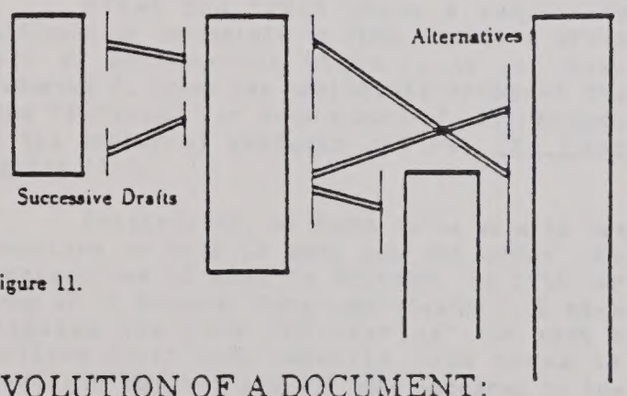
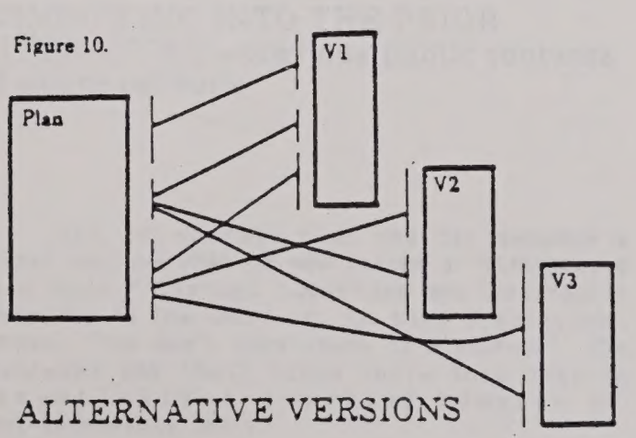


Figure 11.

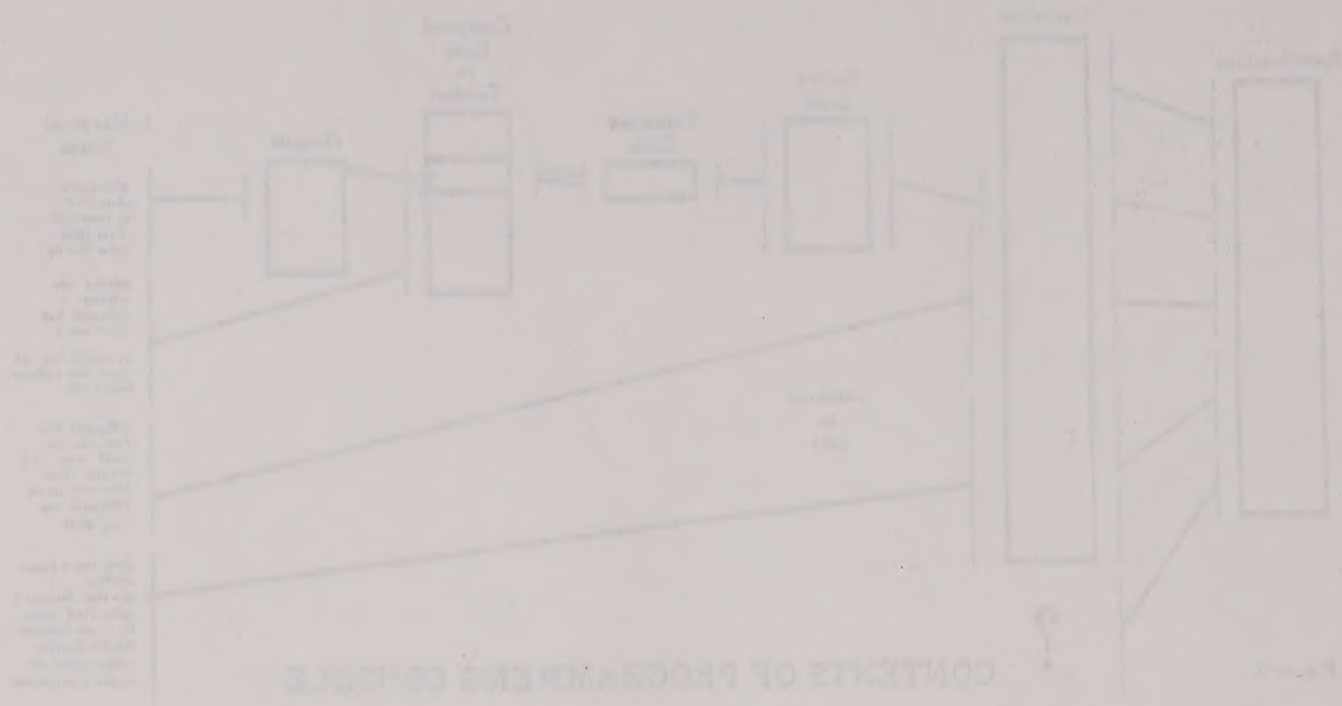
EVOLUTION OF A DOCUMENT:
equilink persists across changes and alternative versions

This link between the same thing in two versions of the same document we call the equilink. Our system automatically keeps track of equilinks when more than one version of a document is shown. The equilink differs from the quote-window in that no occurrence of the section has precedence as the original or owning document. (Conversely, the quote-link may thus be seen as a directional equilink.)

Figure 10.



ALTERNATIVE VERSIONS



The scope and generality of this feature may not be immediately obvious. To many programmers it seems like "too much trouble" to create facilities of this kind; however, now that it exists, it may be used for any casual purpose of the user.

Anyone who has worked with computers is accustomed to the frequent and disagreeable problem of keeping backup files. Many of us also maintain files of previous versions, and/or alternative versions of files for different purposes. These trivial problems create endless annoyance because they are not ordinarily taken care of automatically.

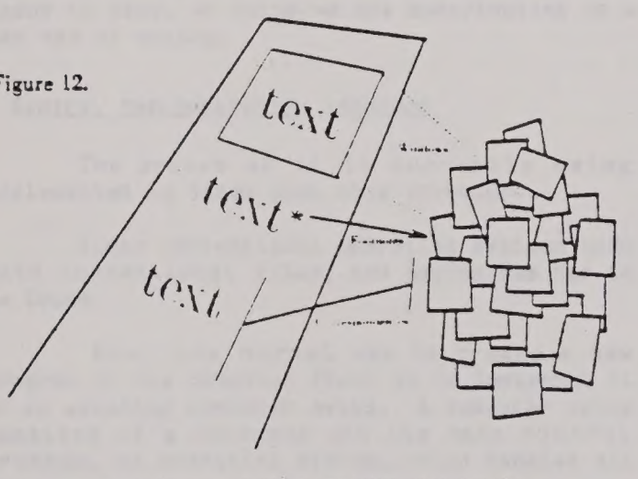
The system described would seem to cut to the center of this problem.

FROM A CONSOLE TO A NETWORK

So far we have more or less assumed a single-processor version of this system, one which easily treats all documents and their versions as an interconnected whole because they are stored in the same place. But given today's network technologies, this is not a real restriction.

The system should be able to grow without size limit, containing in the body of available writings whatever anyone has stored from any place on the network. A user at any location may store what he or she wishes; links may be created by anyone, from anywhere, to bring a document (or part of one) to the inquiring user.

Figure 12.



WINDOWING INTO THE PRIOR LITERATURE—previous public contents of entire network

All of storage near and far becomes a united whole-- what is now called a "distributed data base." Actual locations are essentially invisible to the user; or, in that traditional phrase, "You don't care where it's stored." The documents and their links unite into what is essentially a swirling complex of equi-accessible unity.

MULTI-PERSON USE

For the time being we will ignore the problem of privacy, and assume that all users are freely sharing their work.

Anything stored by one user on the system may be quoted -- adopted into a document-- by another person writing on the system. No copy is made of the quoted material; rather, a quote-link symbol (or its essential equivalent) is placed in the quoting document. This quotation does not affect the integrity or uniqueness of the original document, since no copy is made. Nor does it affect the ownership: in our planned service, a standard proportional copyright fee is paid automatically by the user every time a fragment is summoned.

The use of the special links dramatically simplifies a host of problems.

No copying operations are required among the documents throughout the system, and thus the problems of distributed update, so familiar throughout the computer world, are obviated. Since quoted material only has to reside in its place of origin, and not in the other documents that quote it, other files that quote it are automatically "updated" when its owner changes it.

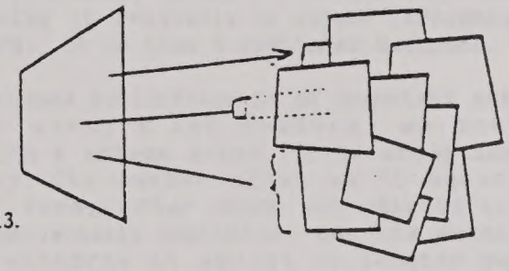


Figure 13.

ANY SUBSET, SUBSTRUCTURE OR GROUPING

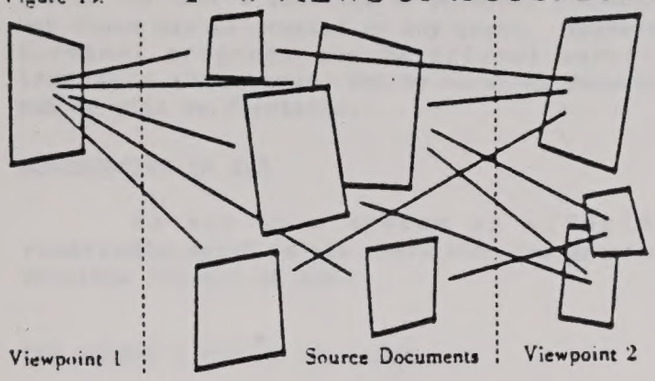
PHILOSOPHY OF THE PERPLEX

Often the truth about a subject is difficult or impossible to find, though a great deal of information about it is on hand. Frederick C. Crews has implicitly proposed the term "perplex" for such a body of information, in his masterful academic satire, The Pooh Perplex (11).

Intuitively, we ought to be able to use computers to help us sort out and order the complexities of what is written, so that our grasp of it becomes firmer and clearer. I have proposed the term "thinkertoy" for such a facility (10); more recently such terms as "decision support system" have appeared in the literature. But what has been less clear is the nature of what such systems should be like.

Our system facilitates multiple interpretations of the same material. Whatever is stored may be seen as a compound object, either organized in different versions or viewed through other documents. Each may independently represent a different point of view. Thus users may highlight different interpretations of the same material, by quoting or linking in from different documents. Thus it seems to be a genuine thinkertoy for strong and subtle intercomparisons.

Figure 11. TWO VIEWS OF A SUBJECT.



We spoke earlier of the unending change of ideas, the way in which a given field is constantly subject to reinterpretation. It is a tradition of western thought that such reinterpretation is always possible, always going on. But how can a literature that has been described in one way be reinterpreted in another, without total rewriting?

Within our system, the user may make marginal notes and new documents that ease his task in totally reinterpreting (or "newly interpreting") the material before him. He has only to make his own summary of another piece of writing-- and indicate the pathway by a correlink. A reader may then summon the corresponding part at any time he wishes to confirm the interpretation.

The improved visualization and control of alternative theories and viewpoints, of disparate and corresponding ideas, should give a person a broader grasp of all everything he reads and thinks about. By providing such tools ready to hand, we think we are contributing to a new way of seeing.

A RADICAL IMPLEMENTATION SEQUENCE

The system as it is currently being implemented is based upon this structure.

Since conventional operating systems work with conventional files, new approaches had to be found.

Now, the normal way to create a new program in the computer field is to implement it on an existing computer setup. A computer setup consists of a computer and its main control program, or operating system, which handles all file access and update. However, since our approach required a radical redefinition of file storage and operations upon what the files contain, the focus of the enterprise has been on the redesign and re-implementation of what most computer people have thought was completely settled, the file system. This in turn requires the creation of a whole new operating system, since the operating systems presently available rest upon a conventional view and implementation of file structures. These have been non-trivial obstacles.

The current version is now in code in a suitable systems language, and we expect to demonstrate the major functions of the system in a reasonable time.

PUBLICATION THROUGH THE SYSTEM ON THE POSTULATED NETWORK

The system's design is a unified whole, but we may think of it as the basic conceptual structure, plus a technical structure which makes it possible, and a contractual structure which making it possible for people to use it confidently. These aspects taken together make a unified design. Because the conceptual structure required very fast lookup within a tightly organized but larged linked system, we had to develop a particular technical structure; and because the conceptual structure expects participants to behave in certain ways, these are embraced in the contract offered to users. These provisions are necessary for the orderly and confident use of published material by many people.

Beyond its use as a private network, we intend that this system be usable as a publication system. Thus a carefully designed system of publication, much like that of paper, has been worked out.

Any user may store anything on the system. Unless specified otherwise it is a private document. If the user chooses to publish it, however, he may do so with relative ease, making it available to anyone throughout the network. It is then a published document.

Because publication is an important act, both for authors and readers, we make publication a solemn event, to be undertaken cautiously. The author signs an "I hereby publish" form, after which not only is the document universally available, but its author may not withdraw it except by lengthy due process. (He may readily publish a superseding document, but the former version remains on the network.)

An author who wishes to render his work universally available, but wishes also to retain the right to withdraw it at any time, has a simple means for so doing. He simply designates his document as a private document with unrestricted distribution. Anyone may have access to it or use it, but the owner is free to withdraw it or change it irrecallably at any time.

Any user may read, or otherwise employ, any published document on the system, or any private document to which he has legitimate access. He can make any kind of links to it from his own documents, private or not.

Accessibility and free linking make a two-sided coin. On the one hand, each user is free to link to anything privately or publicly. By the same token, each other of a published work is relinquishing the right to control links into that work. This relinquishment is part of the publishing contract.

The user may employ any terminal, graphical or printing. Viewing-methods and manipulations are up to the terminal designer. No restraint is contemplated as to what use may be made of the materials found on the system, since no restraint is possible.

We will recommend certain programs for use on the user's terminal or personal computer, but these may be created by any party. Approved terminal programs may be offered certain trademark advantages, but no terminal behavior can or will be forbidden.

NONCOERCIVE IN USE

We see this system as offering remarkable power to all users with the greatest possible freedom of use.

THE SYSTEM'S FUTURE AND SWEEP

We have created this system intending to offer a viable alternative to all forms of reading, writing, archiving and study now handled by methods of paper. Through the system it is possible to mimic, perhaps viably, many aspects of the great society of paper: books, magazines, private notes, copyright, royalty, archiving, and roles for author, publisher and critic.

We want it available to everyone at $2 an hour, and to assure freedom of speech on the system, the integrity of copyright, and other high-minded objectives with respect to its broad future use.

Obviously this is a somewhat ambitious plan, and we cannot judge its viability ourselves. But even given more moderate goals, it seems to be a versatile structure for other purposes. For instance, it would seem to be a good public-access memory service, offering a back end with much greater flexibility than standard storage. Objects can be stored "seamlessly," and the user need have no concern for their size, naming of alternative versions, linkage and the like. Thus it would seem also to be a favorable storage structure for naive-user systems of all kinds, by its removal of various levels of complication.

It should go without saying that the system may be used for all other forms of linked data base, including animated graphics, which we hope will be an important component of future educational and leisure systems.

Such an approach offers to standardize, not languages or processors or algorithms, but the storage form of linkable and complex materials, and terminal interfaces for their exploration. This would seem also to be a worthy goal.

It may be noted that the system adapts readily for purposes of "electronic mail," using the null adaptation.

CONCLUSION

Text is the self-portrait of human thought; more precisely, it is the ordered presentation of the results of that thought. Specific textual conventions have evolved in different aspects of human endeavor, but to study any of them by itself is misleading, like studying only one part of the body or only one sex of a species.

The computer field has gradually become aware of text problems, but most computer people see them as independent areas, like "word processing" and "electronic mail" and "information retrieval."

And so it is that computer people have for the most part never looked at the whole picture. Conventional system designers have approached small subsets of the grand text problem, and the resulting designs have tended to be complicated and cumbersome. Often they require, not only a tangle of specialized programs and user languages, but new social roles for supervisors and service personnel, since ordinary people cannot be taught their use. System complexity appears to rise in proportion to system size, or worse, by some power of system size. But a little thought shows at once that this cannot be permitted. At that rate nothing is going to work.

The view I am suggesting is that the problem is unique, singular and enormous, and the solution can therefore only be unique, singular and enormously simple. I think there are not many text problems but one problem, the text problem, which is the grand interplay of written materials, their interconnections, and the minds that play on these interconnections like harp-strings.

There is a specialty in the computer trade called "system design." Now, we all design systems, but is there a right way? (In some cases a system is intended to do something utterly new, and so there is nothing to be studied or replaced, but that is not germane to our problem.)

By some accounts, system design is the study of existing methods in some area of human endeavor, and the translation of these methods to a computer equivalent of some kind. But this broad description is not very helpful.

To observe and copy is not enough. True, the job of the system designer is in part to observe what people do in an existing system and take note of all the different activities that he sees-- no matter what the people think they are doing or seem to be doing. But the job of the actual design requires more. A designer necessarily makes compressions and adaptations. This is the creative part. And the designer should seek simplicity.

There is no reason, in systems design, to ratify and perpetuate those individual and local complications of life which have arisen in various contexts. Just as the scientist seeks generalizations, unifying ideas which summarize and compress all the varying details he may observe; so the system designer seeks also an underlying structure in whatever existing system he is adapting to the computer. But this structure is not merely empirically found; in part it is imaginatively created, and represents the designer's conceptual encapsulation of what he thinks is going on, and what he thinks should go on.

I suggest, then, that it is the designer's task to find simple structure wherever it may really lurk, and even to create it when he can, though it wasn't there before. This is especially true where non-technical people are going to use the result. Because only truly simple arrangements can be extended, and transposed, and elaborated, by ordinary people in the thoughtful pursuit of their concerns.

Only the simplest of conceptual structures can be extended across all the known functions of text. But we can have simplicity along with power and subtlety; and from extensible kernel structures provide all the functions now available in text systems, and many that are not available but would be more than welcomed. Those are the premises of our design.

This system has been under independent development for twenty years. Its marked departure from other systems and approaches has been greeted with indifference, disbelief or mystification. The only similar system known is Vannevar Bush's hypothetical "memex" system, postulated in 1945, of which we believe this to be the only existing instance. We hope no other will be needed.

While many other features are included in the system as designed for public operation, the central idea is that this conceptual structure (or virtuality) is extensible across the entire range of uses already known for the printed word. What its social effects will be cannot be foretold.

We would like this to be a principal publishing means of the future.

REFERENCES

1. Vannevar Bush, "As We May Think." Atlantic Monthly, July 1945, 101-8.

2. Douglas C. Engelbart, "Toward Integrated, Evolutionary Office Automation Systems." Proceedings of the October 1978 Joint Engineering Management Conference, pp. 63-68.

3. Douglas C. Engelbart, "NLS Teleconferencing Features: The Journal, and Shared-Screen Telephoning." Proceedings of 1975 COMPCON, September 1975.

4. Douglas C. Engelbart, R.W. Watson, and J.C. Norton, "The Augmented Knowledge Workshop." Proceedings of the National Computer Conference, 1973.

5. J.C.R. Licklider, Libraries of the Future. MIT Press, 1965.

6. Frederick W. Lancaster, Toward Paperless Information Systems. Academic Press, 1978.

7. Theodor H. Nelson, "The Hypertext." Proceedings of the World Documentation Federation, 1965.

8. Theodor H. Nelson, "A File Structure for the Complex, the Changing and the Indeterminate." Proceedings of the Association for Computing Machinery, 1965.

9. Theodor H. Nelson, "A Conceptual Framework for Man-Machine Everything." Proceedings of the National Computer Conference, 1974.

10. Theodor H. Nelson, Computer Lib. Published by the author; available from The Distributors, 702 South Michigan, South Bend IN 46618.

11. Frederick C. Crews, The Pooh Perplex: A Freshman Casebook. E.P. Dutton & Co., Inc., 1963.

12. Theodor H. Nelson, Literary Machines (in preparation).

ACKNOWLEDGMENTS

I would like to thank the principal technical contributors to Project Xanadu, especially John V.E. Ridgway, Cal Daniels, William F. Barus, Stuart Greene, Mark Miller, and Roger Gregory. Special thanks are also due to Theodor Holm, Robert W. Fiddler, Jonathan Fagin, Nathaniel Kuhn, John R. Levine, Glenn Babecki and Eric Hill.

© 1980, Theodor H. Nelson

Copyright 1982 by
XOC, Inc.
P.O. Box 7165
Ann Arbor, MI, 48107
(313) 663-3637

XOC, INC.

The Xanadu Hypertext System™

DATABASE TECHNOLOGY

The Xanadu Hypertext System™ is a tool for storing, retrieving, and organizing information. It is not, however, a "Database Management System" in the standard sense of the term. It differs both in its form and in its function.

In the Xanadu Hypertext System™ information is stored in the form of documents: letters, memos, articles, engineering specifications, scribbled notes, textbooks, manuals, computer programs, contracts, reports, and so on. These can vary greatly in their size and in the nature and complexity of their internal structure. This contrasts sharply with database management systems where the information stored consists of aggregates of records holding chunks of data in fairly fixed formats and in fairly homogeneous arrays. In short, Database Management Systems store lists of items, where every item in the list has the same structure. Clearly, these two forms of information storage have different purposes and are appropriate for different kinds of information.

Database Management Systems arose out of traditional record keeping systems in which information was sorted on paper in the form of ledgers, lists, card files, etc.. The storage, retrieval and update functions were performed by hand, often by armies of clerks.

As these functions were computerized, it was discovered that that the computer made possible a variety of capabilities that had not previously been considered. In addition to allowing vastly larger bodies of data to be accessed and manipulated far faster and more accurately than ever before, entirely new functions could be performed. It was found that, if the data was sorted according to certain kinds of generalized schemes, one could impose different patterns of organization on the same body of data, allowing appropriately different organization schemes to be used for different applications. It was also realized that, in addition to storing the data, itself, one could store information about relationships between different chunks of data. Further, it was possible to manipulate this relational information using the very same techniques. One could even store relational information about other relational information, enabling the most general organizational structures to be built.

Today, Database Management Systems are a standard tool for handling information. Research goes on to make them

ever more powerful, efficient, and general-purpose. Tomorrow's Database Management Systems will even more sophisticated.

THE XANADU HYPERTEXT SYSTEM™

Techniques for handling information in the form of documents, however, have not kept pace with database technology. While there are numerous systems available that allow one to store, retrieve, and manipulate to some extent document-type data using a computer, systems that allow one to organize and to interrelate this information (in the sense that Database Management Systems allow one to organize and interrelate bulk data) have been crude to non-existent. The Xanadu Hypertext System™ ends this state of affairs.

The Xanadu Hypertext System™ is the first effective system -- and so far the only effective system -- for document management that allows one to organize and interrelate document type information in a totally general and application independent fashion.

This is what the Xanadu Hypertext System™ can do:

-- It can store, retrieve, and update documents on command.

-- It can create on command explicit linkages among

arbitrary sets of arbitrary portions of arbitrary documents and other linkages. It can retrieve these linkages at will, including information about what is being linked to what and about the nature or type of the link, itself. It can maintain these linkages correctly in the midst of changes to the very data being interlinked.

-- It can maintain multiple versions of any given document, efficiently storing the common portions in common with separate portions of storage required only for those parts of the documents that are actually different (that is, it does not need to store successive versions of a document, only the successive changes to that document). It can provide historical trace back information in dated chronological order about any and all changes to a given document for the purposes of error recovery or auditing.

-- It can do all of this efficiently with nearly linear overhead in terms of both storage space and response time. This means that storage requirements will grow as a constant multiple of the volume of data store (as opposed to many systems of various sorts in which storage overhead increases much faster than the volume of data

does). Response time will remain effectively constant, no matter how vast the body of documents may become. It can make effective use of ultra-high density write-once storage technology, such as optical video disks. It can effectively and efficiently operate in a totally distributed storage environment in which the data is spread over a network of intercommunicating computers that may be located all over the country or all over the world.

This is what the Xanadu Hypertext System™ system can do. No other system is presently available that can do these things. We believe it is quite probable that none will be able to do them for some time to come.

XANADU HYPERTEXT SYSTEM™ OPTIONS

XOC, Inc. possesses a variety of proprietary algorithms and data structures that have already been used to implement a prototype back-end program for the Xanadu Hypertext System™. This system is a powerful, new, and effective means of storing, manipulating, organizing, and retrieving text and other sorts of data.

The Xanadu System™ is suited not only to handling literary data but to managing complex bodies of technical documentation, supporting maintenance of complex pieces of

software, providing the backbone for sophisticated management information control and communications systems, and numerous other advanced applications. In addition to text, it can handle data that is structured in any number of dimensions, including pictures, multidimensional models, cartographic data, and so on.

The Xanadu System™ consists of two major parts: the "back-end" and the "front-end."

The back-end is the heart of the system. It is the part that stores, retrieves, and updates the information being handled.

The front-end is the user interface. It is the part with which users actually interact. It enables the user to view, enter, and manipulate the information that the back-end is handling: it is the go-between, linking the user (a human being) with the back-end (a complex and obscure hardware-software system) via a friendly and easy to use medium.

Both the front-end and back-end are programs executing on their respective computers with appropriate data storage and input/output equipment.

Our back-end design is totally general and, therefore, application independent. This means that the same back-end

program can be used, without significant modification, for all of the information handling applications mentioned above--and many more, besides. We have designed a protocol by which back-ends and front-ends may communicate with each other, enabling users to have full access to the tremendous functional capability of the Xanadu System™ while encapsulating in a single neat package all of the bothersome or proprietary details that remain safely hidden from the user's view.

Front-ends, on the other hand, may be application dependent, for different interactive techniques may be appropriate to different uses of the system. A single, general-purpose front-end will be adaptable to numerous additional applications with only minor modifications. We have produced a skeleton version of such a general-purpose front-end. It is a prototype that serves to demonstrate the capabilities of the back-end.

The range of possible products or services that XOC, Inc. can provide is large and includes, but is not limited to the following:

1. Complete special-purpose software systems for in-house use by the customer: back-end program and front-end program(s) developed for the customer's application(s), to run on the customer's own existing

hardware. Though the following list is by no means exhaustive, some potential applications include:

-- software maintenance
-- engineering project documentation management
-- literary data storage and retrieval
-- cartographic and geographic database management
-- management information services
-- intra-organizational communications systems

2. Complete, special purpose, hardware/software systems for in-house use by the customer for which XOC, Inc. would specify, procure, and sell or lease appropriate hardware.

3. Back-end only software systems for in-house use by the customer: the generalized back-end would be sold, and the customer would develop his own front-ends or procure them from independent vendors. This would be appropriate for customers with highly specialized needs and sophisticated, in-house software development capability. This would also be appropriate for customers with specialized but classified or proprietary applications. In addition, it would be appropriate when independent vendors are selling front-end systems for

our back-end (a situation XOC, Inc. intends to encourage.)

4. Back-end only hardware/software systems for in-house use by the customer for which XOC, Inc. provides the hardware.

5. Front-end only software systems for customers with back-end already installed or who have access to a back-end system (as in situations where customers may wish to expand their installation, add applications, and so on.)

6. Front-end only hardware/software systems for customers back-end already installed or who have access to a back-end system, for which XOC, Inc. provides the hardware.

7. Commercial public access system services at the front-end level: similar to commercial database or time-sharing services except XOC, Inc. would provide a Xanadu System™ to customers rather than a database system or a time-sharing system. Customers would access front-end systems (operated by XOC, Inc.) through ordinary commercial communications services (phone company, Telenet). These front-ends would connect with a back-end or back-ends (also operated by XOC, Inc.). Each

customer would have his own data stored in the system.

8. Commercial public access system services at the back-end level for which the customer would provide the front-end that would connect directly with the XOC, Inc. operated back-end.

9. Public access library and publishing systems services in which information may be "published" on the system and, therefore, made available to all other users. (In #7 and #8 above the emphasis is on commercial data management applications; here the emphasis is is on individuals as users of a kind of sophisticated amalgam of Telex and the public library.)

10. Public access facilities: In conjunction with its library and publishing systems services, XOC, Inc. may wish to provide facilities in the style of a retail sales operation or a fast-food chain where users can come to access the system through on-site hardware. This would enable XOC, Inc. to sell services to users who can not afford to purchase terminals or front-end systems, who may have only an intermittent need for the system, or who may have some special requirements such as the use of special purpose

input/output devices that these facilities would have.

11. Training, consulting, and educational services In conjunction with all of its products and services, XOC, Inc. will be able to provide user support in the form of training and consulting services for the Xanadu System™'s users. This will be in addition to the minimum needed to market the company's products and services.

12. OEM software: back-end software, front-end software, or both, sold to the customer for incorporation into systems the customer puts together for sale to third parties.

13. Enfilade solutions: the techniques used to construct the back-end can be used to create innovative and valuable programs to address a wide range of applications with hitherto unheard of power.

XOC, Inc. would be pleased to provide you a demonstration and to discuss your needs with you. We look forward to serving you.

FIND A FICTITIOUS BUSINESS OWNER V04

ENTER NAME: _____

LINE	FBN NBR	OWNER NAME	BUSINESS NAME
1	1994 163900	NELSON, THEODOR H	THE NETWORK MINT
2	1994 163899	NELSON, THEODOR H.	HYPERMAN
3	1995 164710	NELSON, THEODOR H.	UNI-TED SYSTEMS
4	1995 168032	NELSON, THEODOR H.	TED NELSON STUDIOS
5	1996 170525	NELSON, THEODOR H.	TRANSCOPYRIGHT
6	1996 170526	NELSON, THEODOR H.	DICTRON
7	2001 189785	NELSON, THEODOR H.	PROJECT XANADU
8	2001 189786	NELSON, THEODOR H.	TRANSCOPYRIGHT INSTITUTE
9	2000 185096	NELSON, THOMAS W., GEN. P	POINT REYES VILLA
10	1996 169909	NELSON, WAYNE	MARIN REFRIGERATION SERVI
11	1997 176572	NELSON, WENDY	3PR
12	2002 192544	NELSON, WENDY	AIRSTREAM GEAR
13	2002 192545	NELSON, WENDY	THREE PEAKS RANCH
14	1996 168836	NELSON, WILLIAM R.	PENNZOIL 10 MINUTE OIL CH
15	1996 168839	NELSON, WILLIAM R.	QUICK LUBE OF SAN RAFAEL

ENTER THE LINE NUMBER OF THE OWNER YOU WISH TO SEE: __ & PRESS SE
PRESS "SCROLL FORWARD" TO SEE MORE NAMES

F3 HELP | F4 SELECT | F5 START | F6 DONE | F7 BACK | F8 FORWARD

LEGAL / FBN

&FICTITIOUS BUS. NAMES 07.10.15 CO. CLERK

Jan/Feb 1981.

Memorandum from Ted Nelson to:
Roger Gregory, Development
Mark Crandall, Administration
Sheldon Liederkranz, Finance
Norman Haase and Tim Broadt, Legal Services
Charles Kane, Accounting

PROJECT XANADU: THE NEXT SIX MONTHS.

After twenty years of development work, whose progress has been highly unpredictable, we are finally at a stage where concrete plans can be put on paper for discussion and appraisal. The purpose of this paper is also to ascertain the concrete costs of different computer strategies and legal steps, and to begin to develop pricing figures for our services.

TWO-TIER CORPORATE STRUCTURE

So far the programming work has all been done on promises of later reward (mainly stock), with the result that some 80% of stock has been committed, figures not to be tampered with in good faith. We will refer to this first company as "the rights company," which will hold rights of intellectual property in the Xanadu System (copyright, trade and service mark and trade secret; patents are not considered likely).

In order to take on additional capital while maintaining control, we plan a second corporation to be held 51% by the Rights Company and of which we will sell stock to the public. This we will call "The Operating Company." (We must sell stock in CA and MA, preferably also PA, MI and NY.)

ALTERNATIVES. The Rights Company will be called either The Computopia Corporation (which already exists) or Project Xanadu, Inc. (which doesn't).

LEGAL QUESTION. What is the comparative cost of using the old "Computopia" corporate shell (Delaware), versus beginning a new one?

LEGAL QUESTION. Can we find a formula for "inflatable debentures" which properly acknowledges corporate debts to programmers and other contributors without having a ghastly short-term effect on corporate debt-to-equity ratio?

LEGAL QUESTION. Do such promotions as the Hypercom and Eternity Machines (discussed later) constitute "puffery" which threatens a Reg A public stock offering?

~~DEVELOPMENT, MANAGEMENT~~ AND OPERATION TEAMS

At the present time, the project is split between two locations: Swarthmore, PA (headquarters chosen for various reasons) and Ann Arbor MI (present location of the development team). ~~A third team, the operations~~ team, will come into being when the system goes into steady and commercial operation. The operations team will have to be close to the development group, as continual changes are anticipated. Thus the geographical split, already serious, has sinister potential for factional division in the future, and threatens the coordination and solidarity needed to get the system underway and operational.

PRESENT MANAGEMENT PROBLEMS

1. Getting the system up.

2. Coordination between development and management teams (Ted Nelson, the only member of both, must now travel a great deal, a serious waste of time).

3. Getting up an operational and revenue-producing system, and integrating this operation with continuing development.

SERVICE.

Our principal intended service is the storage and rapid supply of text, graphical and other data, by documents and by links.

Users may be corporations or individuals. They may operate through telephone links, or special hookups (such as laser and microwave) to our Xanadu Stations. Or they may use terminals in the Xanadu Stations on a walk-in basis.

We believe we have unique and sweeping trade secrets, including special knowledge of data structure which obsoletes all generally-known large-scale systems. We have woven some of these tricks into a complete, consistent and broad-scale arrangement for all data. Xanadu service is a precisely defined and uniquely integrated system of storage and retrieval, offering:

1) instantaneous access to any part of any version of any document.

2) arbitrary links within and between documents.

3) historical backtrack.

4) alternative versioning.

In addition to text, we offer unique control of large graphical data bases (such as Landsat photographs or ship blueprints). Marginal notes and other private links may at any time be made on anything stored (text, graphics, music, blueprints.) This facility alone is unlike anything else anywhere.

The Xanadu Publication Service will offer to publishers (who may themselves be authors) a unique new form of on-line publishing. In addition to instant accessibility and automatic royalties (as offered, e.g. by Micronet), our linkage and windowing facilities permit hypertext and compound documents to any remove.

THE STATIONS

Xanadu service will be performed at local franchised stands, or Xanadu Stations (the name of the station will ordinarily be the place's name, as in "Xanadu Station Swarthmore").

The station is the place of basic Xanadu service; it is a place of storage; it is a social center; and it is the place of customer induction, where a new customer is welcomed and shown how to use the system by appealing personnel.

An individual user will ordinarily connect to his local stand. All materials stored within the entire Xanadu net will thereupon be instantly available to him.

The stations will feature large-scale disk storage (eventually hundreds of spindles) and the computer power necessary to serve the local district and process remote requests.

It is important that the first Xanadu installation be not just a faceless over-the-wire service, but actually have a walk-in location with terminals and employees. This will be needed as a continuing focus of interaction with the public and of market research, learning people's attitudes as they use our systems and finding out how best to present them. Thus from the beginning we must have a trained, appealing and intelligent operations crew at the first location.

At the outset we will, of course, use available buildings. At a later time, however, we envision the Xanadu Station as a gleaming stainless-steel modular building of tubular rooms (hexagonal viewed from above), growing taller and wider as its business expands. The special hexagonal architecture will incorporate both seating and terminal wiring; a growing geodesic dome at the center will house a greenhouse lounge and social area.

Eventually we plan to staff these stands with a mix of local people and a circulating corps of Xanadu people. This circulating corps will be something like airline stewardesses and stewards in outlook: oriented to the system as a national entity, moving from stand to stand around the country. They will be intelligent and widely read on the system; they will be able to show others material of interest in many areas, and explain it both to beginners and laymen.

Initially users will pay at the desk by time elapsed. Later, however, we will probably have automatic ticket-vending using an "Addfare" mechanism like that in the D.C. Metro.

Revenue at the Xanadu stations will be provided by standard service, terminal rental, snack bars (with cart delivery of refreshments to your terminal), printout services, picture input scanning, mail services to other networks, and the sale of equipment, souvenirs and trinkets.

ELECTRONIC MAIL

An auxiliary service we can easily add may be an early revenue producer. This is "electronic mail," the increasingly-popular interchange of written notes between computer users. It has the advantages of speed (near instantaneous), convenience (the receiver answers when convenient-- unlike the telephone), and, in some cases, automatic filing.

To hook up the Xanadu system for electronic mail, we would merely have to arrange with GTE Telenet , the main digital carrier, to allow messages to "So-and-so at Xanadu" to be sent into our computers.

The Xanadu net could then serve as an easily-obtained "prestige address" for messages, requiring no other membership or eligibility, as do most other electronic mail systems. Thus

CENTAUR@XANADU

becomes a sufficient electronic mailing address from anywhere in the world.

Our special advantage for electronic mail, even for users of other nets, is automatic filing: if you want to save something that has been sent you, you don't have to put it away-- it's already there. Whether sender or receiver pays for storage, however, still has a few undecided problems.

PRODUCTS

1. The In-House "Data Engine."

We are of course capable of leasing the software for an entire free-standing system within (say) a publishing firm, a legal firm or the White House. The currently fashionable term is "data base machine." Ours we would call The Xanadu Data Engine(tm).

Dangers. This carries with it the unfortunate vulnerability of our programs to copying, intense study and scrutiny. We foresee no danger of the copied program being openly sold, since its origin could easily be proven. Neither is there any real danger of changed or transposed versions appearing, since to change it you must understand it thoroughly.

The greater danger, however, is that our software could be gradually pried apart and its secrets ascertained by an extremely good programming team. (For instance, the group at Brown University, who have been appropriating Nelson's stuff for twelve years.) This in turn could greatly weaken our competitive position.

It must be assumed that once the program is leased, it is "gone," and that copies of it will be passed around sub rosa thereafter. On the good side, however, we do not believe that the deeper secrets could be ascertained for a considerable length of time-- say, two years. It can also be argued that releasing the program will be a better way to distract the competition than not releasing the program, which would encourage others to start thinking it out from first principles-- as we did, successfully.

POLICY QUESTION: SHOULD WE LEASE THE SYSTEM, AND UNDER WHAT CIRCUMSTANCES? (IF SO, SHOULD WE TAKE TIME TO RIG IT WITH CONCEPTUAL BOOBY-TRAPS?)

LEGAL QUESTION: WILL NON-DISCLOSURE AGREEMENTS HELP, AND IN WHAT FORM?

2. Front End Programs.

Basic Xanadu service, as provided by the stations, must communicate with an appropriately programmed display computer at the user's command. This local machine we call the Xanadu Front End or FREND(tm).

While we will encourage software vendors to create these front-end programs, we will also offer our own for some machines. One is presently under development for the Heath/Zenith terminal computer.

DEVELOPMENT SCHEDULE

Anyone with experience in High Software knows that programming cannot be scheduled in any usual sense. We can, however, specify the order in which work is to be done.

Phase One is what it takes before we can say, "the system is up." This includes:

Storage, retrieval of text; all editing operations
Arbitrary links, including windows
Multiple and alternative versions
Slow historical backtrack
Basic front-end program for Heath/Zenith machine
Issuance of front-end specifications to other programmers and vendors

At this point the system can be used and even generate revenue. (Unfortunately, if leased, it is also more vulnerable to prying than it would be at a later stage.)

Phase Two involves a number of other fundamental and necessary features; when all are completed, the system will be capable of full operation.

Royalty (and billing)
Graphics (bit-map)
Graphics (sparse, vector and coded)
Fast historical backtrack
Security and private documents
Networking or "galactic structure"

Naturally, further enhancements will continually be desired and required (such as concurrent concordances, special data structure compiler) but this is the main Xanadu system, and will allow us to open the first station.

POLICY QUESTION: should we go for cash flow with any of the Phase Two features missing?

HIGH ROAD, LOW ROAD

There are two major alternatives for getting the system up in the six-month future. Since our language is "C," we must use computers offering C-- principally the Vax and the Onyx, but possibly others.

1. The High Road involves obtaining a VAX computer, capable of serving several hundred users, and setting it up as our first Xanadu station. (In addition, we would hook it to Telenet, making Xanadu service at once available to Telenet users.)

The cost of the machine, with about a gigabyte of disk storage, would be approximately half a million. This is probably available on credit. After that, however, we need cash flow to keep up the payments, so such a move may not necessarily be advisable.

2. The Low Road is to get one or more additional Onyx computers (at $20,000 apiece) and offer remote service on them. This makes disk storage marginally more expensive but permits expansion by small increments as demand increases reliably.

Intermediate Roads would involve other machines offering C, such as the Prime or the PDP-11 series.

FIGURES TO DEVELOP: COMPARATIVE COSTING OF DIFFERENT MACHINES.

Note that only approximate performance figures can be obtained. Only when the system runs on a given computer will we know how many high-performance users it will support. (Example: we expect the Vax to support at least a hundred simultaneous high-performance users, but the figure might reach as high as 500.)

Another way to look at it: what level of service do we deem to be "acceptable" performance? As the number of users increases, so do the delays experienced by all. As we add users to gouge more money, what level of diminishing performance will the users accept?

PRICING.

(Note the problem of pricing to avoid saturation; we could be profitable at quite low rates.)

We have as yet no firm prices. Here are the principal prices to be established, with very tentative opening figures.

FULL SERVICE (connection and all disk fetches)

1200 baud / $20/hr prime time, $5/hr night
300 baud ~~$7/hr prime time, $2/hr night~~

(Communication charges, such as Telenet, are additional.)

DISK STORAGE (PUBLISHER MUST GUARANTEE AT LEAST 1 YEAR DISK STORAGE)

$100/megabyte-month

ROYALTY TO AUTHOR or AUTHOR'S FUND

10% deducted from full-service charge.

TAPE FETCHES of material not on disk

To open a document, .25; .02/kilobyte thereafter

TERMINAL RENTAL AT STAND

$2 to $10/hour, depending on equipment.

PUBLIC RELATIONS

In the first six months of 1980, four major public-relations thrusts will occur:

1. Continuing publicity to the hobby and personal market, including issuance of user-names ("Xandles"), Xanadu Credit Cards, and various propaganda articles in personal-computer magazines.

2. Hypercon Six ("Stargate"). The sixth of our semi-annual hypertext meetings, tentatively scheduled at Princeton, April 11. Our lead speaker is none other than Alan Kay of Xerox Palo Alto Research Center, developer of the Smalltalk language and considered by many the world's greatest programmer, who has tentatively said he could attend.

3. Literary Machines, the official book of the Xanadu System, describing the system in ruthless detail, its history, its ideals, the competition, the economic and political problems. Our proprietary secrets will of course not be revealed. Chapters will be placed in major magazines (Datamation already assured.) June 1 is tentative publication date.

4. Viewdata 81, the international Viewdata Conference (Toronto, May). TN will give talk: "Why Not Have It All?"

MARKETING

Whether we can find early customers willing to commit to our service at such high costs as $20 an hour is not clear. Some key targets are obvious:

1. ENGINEERING. Our alternative-versions facility offers unprecedented control of large-scale project designs (such as different model 747 aircraft). This requires, however, very savvy customers and considerable additional programming effort (3D graphics and front-end functions).

2. DOCUMENT CONTROL AND DISTRIBUTION. There are numerous applications where complex documents have to be constantly regenerated out of pools of common material,and at least the latest version immediately accessed by various workers in the firm. Such industries include:

PUBLISHING HOUSES
LEGAL FIRMS
PROGRAMMING HOUSES
THEATER
PUBLIC RELATIONS
GOVERNMENT
FOUNDATIONS

Our strength comes from reconstituting each document on demand from a common bank of material. Cross-indexing, since it takes place within that pool of contents, involves no complications of the other methods.

However: getting this across to organizations entrenched in a low-level "word processing" outlook may be difficult indeed.

POLITICALLY SENSITIVE MARKETS

1. Viewdata or "Teletext" or "Videotex."

Text systems which use the home TV and an "adapter" (actually a computer terminal) are being promoted in England, France, Germany and Canada.

The Xanadu system could actually become just such a system-- far superior, we believe, to these others-- with no adaptation. This is, however, a touchy political matter.

Australia is already interested in our system, and we intend to demonstrate at the Viewdata conference in Toronto in May.

US Viewdata standardization is unlikely; thus we would have to negotiate through cable companies. Large amounts of lobbying money would probably be needed, however.

2. Government. National Science Foundation has asked for an in-house Xanadu system. Other key placements (such as Congress, Executive Office of President) would not be hard. Fraught with both lobbying/sales and program-security problems, however.

APPENDIX. Proposed Contracts.

USER CONTRACT.

The user desiring services on credit may sign a contract/credit request. Choices offered may include:

monthly, bimonthly or quarterly billing
additional cards or subaccounts for employees/family members
cost ceilings on subaccounts
food-and-trinkets/services-only subaccounts
registration of one or more "Xandles," or user-names
registration of public-key encryption numbers
"gold card" Xanadu membership, allowing you to shop for and charge expensive objects through the system

A key feature of the contract will involve automatic archiving to tape, with or without encryption. It is assumed that the typical user wants permanent safety for his files, but if they are archived it will be in an indelible form, sandwiched between many other archived materials. This service must be explicitly signed for to prevent lawsuits at a later time mandating the destruction of archives. "If you sign for the full service, you can't make us throw away what you said to keep-- but only you can get at it."

The problem also of access by heirs, literary executors, etc., must also be studied from a legal point of view.

PUBLICATION CONTRACT.

The publisher of a document may of course be anyone. He signs a contract (on a credit-card-type form) agreeing:

To pay a minimum of one year's disk rental for the material stored.
To accept our standard royalty, whatever that may be, and our accounting of it, for all use of the materials.
To allow other Xanadu users to link and window to these materials, provided only that he gets the royalty whenever materials from the specified piece are summoned.
To give six months' notification of intent to depublish.

That he is the proper and legal copyright holder.
That the material does not violate national security, that its use does not violate any laws, that its use does not infringe on copyright, trade secret, or other forms of intellectual or other property.
That it does not contain libelous material.
That he will be liable for any tortious or illegal contents.
That materials may be liable to removal if found to be in violation of such provisions as ~~the above~~, and to removal on peremptory challenge by affected parties if the publisher, he, cannot be found.

NOTE: The Xanadu system wants to place itself in the position of a printer, rather than a publisher, and ~~in addition~~ to distance itself wherever possible from any involvement with the content. (The legal doctrine that a printer may be liable in cases of libel is actually quite dangerous to us.)

FRANCHISE CONTRACT.

The contract should be very like the McDonald's contract.

The franchisee agrees to work in the station himself and conduct its business according to our rules, using buildings, software, advertising, decorations and costumes we supply, taking on circulating National Personnel, etc.

The discovery of links to organized crime or intelligence agencies is grounds for termination.

The franchisee agrees not to reveal our trade secrets or allow copies of our program to fall into unauthorized hands. All employees, too, must ~~sign non-disclosure~~ agreements.

NODE LEASE CONTRACT.

The touchiest contract will govern the lease of the Xanadu program for in-house use.

If the program is so marketed, we will need some basis for barring customers that we don't trust. In addition, the contract must include:

Non-disclosure agreements for all employees who run the computer.

Agreement not to offer services to the public in competition with our Xanadu stations.

Cancellability by us at the close of a specifiable term of time.

Our right to mandate specific program changes from time to time at our pleasure.

TRADEMARK CONTRACT FOR FRONT-END PROGRAM

The developer of a Xanadu front-end program to connect to the Xanadu system from a user's computer (Apple, Wang etc.) will be granted the trademarks "Xanadu FREND," and others, for a certain period of time in return for:

Meeting our specifications as of such-and-such a date.

Small payment (one to five dollars a copy)

Agreed-upon tasteful display of trademark, etc.

It is to be understood that the trademark specifications will be under continual development, and front-end programs which are brought up to specifications as they change will be allowed to keep the trademarks on a continuing basis.

[for National Computer Conference, 1977] 12 Jan 77

DESIGN OF A TRANSCENDENTAL LITERARY NETWORK

Theodor H. Nelson
Department of Mathematics
Swarthmore College
Swarthmore, PA 19081

For personal learning and recreational reading, for various kinds of in-depth research, ~~as well as~~ and for historical and background study in many fields, we need full-text access in great quantities.

The computer can contribute here, not by digestion and summarization, as in some forms of retrieval, but by extremely quick full-text delivery, and especially by certain sorts of cross-access.

The network proposed here will hold an extraordinarily large corpus of documents for rapid viewing, and for extremely fast cross-fetching along links between them. These links, put in freely by authors and annotators, may be freely used by the reader as paths leading to other documents.

The types of links proposed here carry out automatically certain lookups which have long existed implicitly in ordinary literature: the jump-link (corresponding to the conventional footnote and citation); collateration (corresponding to side-by-side printing and marginal notes); and quote-windowing, whereby the original context of a quotation may be summoned. While other linkage modes are possible-- notably annotated

and "colored" links-- this set has the advantages of simplicity, recursive extensibility and psychological clarity over many removes.

As these linkage facilities extend the notion of literature considerably, we use here the term "transcendental literature" for a corpus of this type, available in this delivery system.

The characteristics of document and authorship are extended to promote orderliness. A "document" may consist of text, links, or both; it has an author or owner. Thus there is no ambiguity of ownership, or danger of the sullying of a work's integrity by someone who is not its author. However, any author may freely annotate or create links to the work of another; these contributions becoming his own "document."

Rapid text-editing facilities are an important feature of the system, as are the storage (if desired) of successive drafts and modifications, whose canonical archival form permits point-by-point comparison among successor versions. Simple extensions will permit string-search and concurrent thesaurus maintenance.

Extensions to graphics and animation are contemplated.

Tele-conferencing capability is of course implicit , enhanced by the ability to link into the greater stored corpus.

Because this is a special-purpose multi-user delivery system, employing minicomputers in a network, low prices are projected: two to three dollars an hour including author

royalties and all basic services, exclusive of communication costs. However, user-written programs cannot be accommodated.

As vital as the technical aspects of the system are its planned accommodations to a free-enterprise economy: notably, that the condition of fast accessibility must be paid for by someone-- called a "publisher"-- and network nodes must be financed through the free-enterprise mechanism, possibly by franchise. Finally, copyrights and royalties are handled automatically within the system.

It is often carelessly asserted that no such rapid full-text delivery system will be useful until it contains "the entire Library of Congress." Quite the contrary: client groups with special interests may justify their own usage for a few million characters of complicatedly linked material.

Most important are the issues of privacy and freedom of speech and the press. Safeguarding these in the ever-growing corpus of our civilization has increasingly technical aspects, but is not a technical problem.